UNIVERSITÀ
DI PAVIA

University of Pavia
DEPARTMENT OF ELECTRICAL, COMPUTER AND BIOMEDICAL ENGINEERING

PhD thesis in Electronics, Computer Science and Electrical engineering

---

Multimodal human-robot interaction strategies in human-robot collaborative tasks

PhD Candidate
Yevheniy Dmytriyev

Supervisor:
prof. Hermes Giberti

Co-supervisor:
prof. Marco Carnevale

Academic year 2022/2023

Ringraziamenti

Desidero dedicare alcune parole di gratitudine a coloro che mi hanno permesso di raggiungere vette che da solo non avrei neanche immaginato. Questo dottorato rappresenta una vittoria collettiva, un risultato ottenuto grazie al supporto familiare, al lavoro di squadra e all'amicizia, superando di gran lunga il valore di un semplice titolo accademico.

Innanzitutto, vorrei rivolgere un profondo e tenero ringraziamento alla mia adorata moglie, Fiorella. Sei stata la mia luce guida, il mio sostegno incrollabile, e il mio rifugio sicuro lungo tutto il percorso di ricerca accademica. Il tuo amore, la tua gentilezza e la tua pazienza infinita hanno reso ogni sfida più leggera, ogni vittoria più dolce e ogni giorno più luminoso. Grazie per essere stata al mio fianco, condividendo con me ogni passo di questo straordinario viaggio.

Ho la grandiosa fortuna di avere al mio fianco due persone che da sempre credono fermamente in me. Voglio quindi dedicare un sentito ringraziamento ai miei genitori, che con il loro amore e il loro sostegno incondizionato rendono possibile ogni mio successo da ormai una vita. Il merito dei miei traguardi si rispecchia nella vostra dedizione, valori e sacrifici. Spero che sappiate che sono orgoglioso di voi almeno quanto voi lo siete di me. Un ringraziamento speciale lo dedico mio fratello Maksym e alla mia sorellina Giulia. Il mio impegno è anche merito vostro. Ho sempre lavorato nella speranza di essere un riferimento e mi auguro che sappiate di poter contare su di me, sempre.

Oltre alla mia famiglia, voglio ringraziare coloro che hanno condiviso con me questa avventura accademica. Ho tanto apprezzato la collaborazione con Ahmed Z., che non solo è stato un collega ma anche un amico, un esempio di determinazione e dedizione ai propri obiettivi personali e accademici. Ringrazio Carol, Federico e Ahmed F., che in questi anni hanno contribuito a creare un ambiente di lavoro stimolante e sono stati un costante sostegno verso il raggiungimento dei migliori risultati. Grazie a Peppe e José e alla loro straordinaria energia che hanno saputo infondere nell'ambiente di lavoro, contribuendo così a mantenerlo dinamico e pieno di nuovi stimoli.

Infine, voglio ringraziare il prof. H. Giberti, la cui saggezza e visione mi hanno guidato verso il pieno sviluppo del mio potenziale nel corso di questi anni. Un sentito ringraziamento al prof. M. Carnevale per avermi trasmesso metodo e rigore, fondamentali per le nostre pubblicazioni accademiche. Inoltre, vorrei rivolgere un caloroso apprezzamento alla prof. V. Furlan, la cui passione ed energia hanno contribuito in modo significativo a mantenere salda l'unità e la coesione del nostro gruppo di ricerca.

Abstract

Collaborative robots, or cobots, have emerged as a transformative solution to address the limitations of traditional industrial robots, with a focus on enhancing production in dynamic environments while ensuring human safety. This doctoral research delves into the realm of collaborative robotics, striving to offer a comprehensive set of guidelines and innovative solutions for the efficient deployment of cobots within the context of the human-centric Industry 5.0 paradigm. The core of this work involves the exploration of human-robot collaboration (HRC), particularly focusing on human-robot interaction (HRI) strategies and associated technologies. The cornerstone is the implementation of advanced human-robot interface devices that make robots more accessible for human interaction and flexible deployment.

This study recognizes the increasing importance of collaborative robotic cells, particularly in improving operational adaptability without sacrificing safety. To automate trajectory planning in contact-based applications, a representative industrial case of the footwear sector is considered, featuring a 3D stereo depth camera, a six-axis collaborative robot, and a force/torque sensor. Experimental validation occurs in the context of glue deposition for shoe manufacturing, considering curvilinear non-flat workpieces. On the other hand, to present novel techniques for information exchange between humans and robots, the integration of brain-computer interface (BCI) technology and the hand-guiding mode into a collaborative robotic assembly task is explored as an exemplary application. A task-oriented algorithm, based on biosignals collected by the BCI, is proposed to provide command messages to the robotic manipulator, while a force/torque sensor can enable the provision of guidance messages, allowing for hand-guiding interaction. Experimental validation demonstrates substantial reductions in average cycle times and improved consistency with respect to a fully manual assembly process, illustrating the viability of this technology for Industry 5.0 applications.

These innovations seek to improve rather than replace the skills of artisan professionals, fortify local industries, and foster innovation. A notable outcome is the exposition of the need for continued refinement in cobot cell design and human-robot interaction, thus setting the stage for further research and design endeavors. However, this work seeks to go beyond the usual technological issues and bridge the gap between the potential of cobots and their actual use in real-world automation. A pivotal contribution, in this sense, is the first-hand account of the application of a visual programming language to program a cobot for an authentic industrial application. This unravels the intricacies of the programming process and the design considerations inherent in the crafting of a robotic cell. A critical analysis of the resultant program illuminates the challenges and merits associated with adopting a simplified robotic programming paradigm. Additionally, to allow workers to increase their creativity, problem solving and managerial skills by using intelligent robotic tools, the proposed alternative layout of the robotic cell takes advantage of the human-centric philosophy of Industry 5.0, to promote flexibility in unloading operations in an end-of-line industrial process.

The technical foundations of the proposed solutions and the corresponding hardware and software tools are presented. Robot program development occurs primarily within the proprietary TMflow environment of the considered robotic manipulator, in synergy with the Python programming language to harness open source libraries for sensor signal analysis. The integration of BCI technology, enabled by a headset capable of collecting and transmitting brain signals, underpins the collaborative human-robot task. Extensive laboratory tests, feasibility studies, and experimental campaigns are conducted, incorporating the robot arm equipped with vision systems, sensors, and end effectors. Simulation and offline programming, facilitated by RoboDK software, accelerate industrial cell design, sizing, movement testing, and exploration of various robot work cell scenarios.

In conclusion, this doctoral thesis explores the world of collaborative robotics, going beyond its initial industrial uses to push towards a more human-centered collaborative cells. It underscores the symbiotic relationship between artisanal craftsmanship and automation, while championing the seamless integration of advanced interfaces into collaborative workstations. In-depth studies have been conducted on a variety of topics related to collaborative robots and human-robot interaction in industrial settings, providing strategies for the successful implementation of cobots in the Industry 5.0 environment. Examples of applications and industrial use cases are explored in the domain of glue deposition in the footwear industry, assembly tasks, and industrial end-of-line unloading processes. The proposed solutions prioritize simplicity, reprogrammability, and modularity in robotic tasks, empowering human operators to assume the supervisory, decision-making, and problem-solving roles. This approach not only increases flexibility, but also reduces the response to market times, making manufacturing processes more agile, adaptable, and competitive in an ever-evolving market landscape.

Sommario

I robot collaborativi, o cobot, sono emersi come una soluzione trasformativa per affrontare le limitazioni dei tradizionali robot industriali, con l'obiettivo di migliorare la produzione in ambienti dinamici garantendo al contempo la sicurezza umana. Questa ricerca di dottorato esplora il mondo della robotica collaborativa, cercando di offrire un insieme completo di linee guida e soluzioni innovative per l'efficiente implementazione dei cobot nel contesto del paradigma dell'Industria 5.0 centrata sull'essere umano. Il nucleo di questo lavoro riguarda l'esplorazione della collaborazione uomo-robot (HRC), con particolare attenzione alle strategie di interazione (HRI) e alle tecnologie associate. Il pilastro centrale è l'implementazione di dispositivi avanzati di interfaccia uomo-macchina che rendano i robot più accessibili all'interazione umana e implementabili in modo flessibile.

Questo studio riconosce l'importanza crescente delle celle robotiche collaborative, in particolare nel migliorare l'adattabilità operativa senza compromettere la sicurezza. Per automatizzare la pianificazione della traiettoria nelle applicazioni basate sul contatto, viene considerato un caso industriale rappresentativo del settore calzaturiero. La soluzione proposta è caratterizzata da una telecamera stereo 3D, un robot collaborativo a sei assi e un sensore di forza/coppia. La validazione sperimentale avviene nel contesto della deposizione di colla per la produzione di scarpe, considerando superfici non planari degli oggetti curvilinei considerati. D'altra parte, per investigare innovative tecniche di scambio di informazioni tra persone e robot, viene esplorata l'integrazione della interfaccia cervello-computer (BCI) e la modalità di guida manuale in un compito di assemblaggio robotico collaborativo come applicazione esemplificativa. Viene proposto un algoritmo basato sui biosignali raccolti dalla BCI, per fornire messaggi di comando al manipolatore robotico, mentre un sensore di forza/coppia permette lo scambio di messaggi aptici, consentendo il posizionamento guidato dall'interazione manuale. La validazione sperimentale dimostra una significativa riduzione dei tempi medi del ciclo e maggiore coerenza rispetto a un processo di assemblaggio completamente manuale, illustrando l'applicabilità di questa tecnologia per le applicazioni dell'Industria 5.0.

Queste innovazioni sono volte a migliorare piuttosto che sostituire le competenze dei professionisti artigiani, rafforzare le industrie locali e promuovere l'innovazione. Un risultato rilevante è l'esposizione della necessità di un continuo perfezionamento della progettazione delle celle dei cobot e l'interazione uomo-robot, ponendo così le basi per ulteriori attività di ricerca e progettazione. Tuttavia, questa tesi si prefigge di andare oltre le solite problematiche di adozione di soluzioni tecnologiche e di colmare il divario tra il potenziale dei cobot e il loro effettivo utilizzo in impianti industriali reali. Un contributo fondamentale, in questo senso, è l'esperienza diretta dell'implementazione di un linguaggio di programmazione visiva per programmare un cobot per una reale applicazione industriale. Questo mostra le complessità del processo di programmazione e le considerazioni di progettazione intrinseche alla creazione di una cella robotica. Un'analisi critica del programma risultante illustra le sfide e i meriti associati all'adozione di una modalità semplificata di programmazione robotica. Inoltre, per consentire ai lavoratori di aumentare la loro creatività, capacità di risoluzione dei problemi e competenze manageriali utilizzando strumenti robotici intelligenti, viene proposto un layout alternativo della cella robotica, la quale sfrutta la filosofia della centralità dell'uomo dell'Industria 5.0, per promuovere la flessibilità nelle operazioni di scarico alla fine di una linea di produzione industriale.

Vengono presentate le basi tecniche delle soluzioni proposte e gli strumenti hardware e software corrispondenti. Lo sviluppo del programma del robot avviene principalmente all'interno dell'ambiente proprietario TMflow del manipolatore robotico considerato, in sinergia con il linguaggio di programmazione Python per sfruttare librerie open source per l'analisi dei segnali dei sensori. L'integrazione della tecnologia BCI, resa possibile dall'uso di un caschetto neurale in grado di raccogliere e trasmettere segnali cerebrali, è alla base del compito di assemblaggio collaborativo proposto tra uomo e robot. Sono stati condotti vari test di laboratorio, studi di fattibilità e campagne sperimentali, integrando il braccio robotico dotato di sistemi di visione, sensori ed end effector. La simulazione e la programmazione offline, facilitata dal software RoboDK, hanno accelerato la progettazione della cella industriale, il dimensionamento, i test di movimento e l'esplorazione di vari scenari delle celle robotiche studiate.

In conclusione, questa tesi di dottorato esplora il mondo della robotica collaborativa, andando oltre i suoi utilizzi industriali iniziali per spingere verso celle collaborative più orientate all'essere umano. Gli studi sottolineano la possibile relazione simbiotica tra l'artigianato e l'automazione, promuovendo nel contempo l'integrazione di interfacce avanzate nelle postazioni di lavoro collaborative. Sono stati condotti studi approfonditi su una serie di argomenti legati ai robot collaborativi e all'interazione uomo-robot in ambienti industriali, fornendo strategie per l'implementazione dei cobot nell'ambiente dell'Industria 5.0. Vengono esplorate applicazioni ed esempi di casi di utilizzo industriali nel settore della deposizione di colla nell'industria calzaturiera, nelle attività di assemblaggio e nei processi industriali di scarico di fine linea. Le soluzioni proposte promuovono la semplicità, la riprogrammabilità e la modularità nelle attività robotiche, consentendo agli operatori umani di assumere ruoli di supervisione, decisionali e di risoluzione dei problemi. Questo approccio aumenta non solo la flessibilità, ma riduce anche i tempi di risposta al mercato, rendendo i processi di produzione più agili, adattabili e competitivi in un mercato in continua evoluzione.

Contents

List of Figures

List of Tables

Introduction

Since the 1960s, the installation of industrial robots in manufacturing plants has revolutionized significantly various industries and sectors, bringing extreme changes in many manufacturing processes. Robotics technology was initially developed for use in the automotive and heavy industries due to the need to automate laborious, physically strenuous, and ergonomically difficult tasks that could be hazardous to human workers. Industrial robots have demonstrated their value in terms of precision, resulting in improved product quality, reduced waste, and increased production. Nevertheless, the incorporation of robots into production systems requires a substantial investment in time and money. Expert programming, simulations, and development are essential to guarantee precise integration. Traditional production sites have augmented their flexibility, but this comes with a hefty price tag, making them only cost-effective for large-scale production companies. Moreover, industrial robots are not suitable for dynamic and unstructured environments, such as those found in small and medium enterprises (SMEs). In these businesses, the expertise of an operator, as well as the autonomy, agility, and flexibility of human workers, are essential. These requirements go beyond the characteristics that a traditional industrial robot can offer. Therefore, the robotic sector is adapting to meet these needs, transitioning to a more collaborative robotics, bringing together humans and robots to work collaboratively. Such synergy aims to be used in complex environments in a straightforward and efficient manner, especially if coupled with other technological advances of the digital era. This evolution of the robotic sector will allow SMEs to benefit from the latest automation technologies without sacrificing the human contribution to the production process.

In recent years, the adoption of Industry 4.0 technologies and the emergence of Industry 5.0 paradigm have led to a search for a shift toward more human-centered industrial processes. The focus moved from increasing the performance of automated production systems to enabling safe interaction between robots and human operators, using human input to optimize robotic cells and execute tasks that would be impossible to fully automate without human management. To meet the requirements of this new trend, technologies that allow for easy and flexible task planning and programming by non-expert operators are essential. Collaborative robots (cobots) have been introduced to address these challenges, overcome some of the limitations of industrial robots, improve production in dynamic environments, and interact safely with human operators. Despite the fact that technology is necessary, it is not enough to tackle all the difficulties that come with the creation and execution of more human-oriented automation. Discussions about collaborative robots often revolve around safety concerns, given the coexistence of mobile machines and humans.

Ensuring safety is paramount for the potential benefits of collaborative robots in the workforce. Continuous risk assessment and management are emerging fields, and effective communication between humans, cobots, robots, and automated processes is vital. Policies and processes should be flexible and effective to address complex challenges.

Although the industry exhibits great potential to increase its efficiency through the use of digital technology and automation, manual labor is still frequently utilized for specialized and intricate physical tasks. The extreme flexibility required from handicraft professionals and more creative jobs is now the new limit for cobots, while the productivity and the efficiency of automated robotic cells may be perceived as a threat to craftsmanship.
The term "Luddite" has come to refer to those who are opposed to new technology. This is rooted in the 19th century labor movement known as the Luddites. These were highly skilled and well-paid workers who were concerned about the effects of new technologies on their craftsmanship. They were unhappy with the decrease in textile quality, the replacement of skilled labor with less experienced workers, the introduction of technology without consulting the skilled labor force, and the widening of the power gap between management and labor. Interestingly, these very issues continue to be relevant today. Today, it is essential for robotics professionals, engineers, and user experience designers to thoroughly evaluate and explain the potential risks, drawbacks, and benefits of incorporating robotic systems in both the workplace and consumer markets. This evaluation should encompass crucial questions about its impact on existing employees, the potential need for retraining, safety considerations in human-robot interactions, cybersecurity vulnerabilities, and the potential for bias in robotic decision-making, which could introduce disparities in power dynamics and benefits within the workplace.
To allow both SMEs and artisans to benefit from the Industry 5.0 paradigm and Industry 4.0 enabling technologies, advanced human-robot collaborative strategies are developed to overcome the current limitations of machines by involving workers in the decision-making process. This integration can improve the quality, efficiency, and sustainability of craft processes by connecting them with the needs and possibilities of small businesses. Digital tools can optimize material usage, reduce waste, and standardize production, while handcrafting can add uniqueness, creativity, and emotional value to the final product. In addition, hand skills are essential for the maintenance, repair, and customization of digital machines and tools, as well as for the design and prototyping of new solutions that can address specific needs or contexts. Therefore, a more holistic and adaptive approach must be adopted, which recognizes the complementary nature of handcrafting and automation. This can foster innovation, diversity, and resilience in the craft sector while also contributing to the development of more sustainable and inclusive forms of production and consumption.

Cobots have been developed from their industrial counterparts to be safer, easier to use, more agile, and adaptable, while still being dependable and accurate. Their adoption had such a relevant impact on robotization that they triggered the need for technical specifications of the International Organization for Standardization (ISO) with respect to the existing norms for industrial robots and robotic cells to deal with the role transformation they brought. This transformation redefined the tasks of robots, as well as their interaction with human workers, requiring modifications of risk assessment and safety measures to allow new implementation strategies.

This thesis investigates human-robot interaction techniques and the current use of cobots in an industrial context with the aim of providing a series of guidelines and solutions to efficiently deploy collaborative robots while taking into account the human-centric approach of the industry 5.0 paradigm. This research aims to demonstrate how collaborative robotic tools can augment, rather than substitute, the abilities of existing artisan professions, aiding in the fortification of local industries and the encouragement of innovation. The gap between the potential and actual applications of cobots in real automation scenarios will be investigated, highlighting barriers beyond typical challenges to technology adoption. This would help cobots become useful tools for people who are knowledgeable about how to maximize their potential and allow workers to improve their creativity, problem solving, and management skills by using intelligent tools.

In the following chapter 1 the state-of-the-art synthesis provides contextualization and an overview of related research topics in the field of Human-Robot Collaboration (HRC), focusing on Human-Robot Interaction (HRI) strategies and technologies. Chapter 3 presents the machines and devices utilized in various configurations, reporting their relevant characteristics. The HRI strategies developed and the collaborative tasks tested will be described in the chapter 4. The main findings will be summarized and the proposed solutions will be discussed in detail. The chapter 5 will introduce the industrial applications developed, highlighting the technological issues solved and the results obtained from the implementations. Conclusions and considerations about the presented methods, as well as the findings and future developments of industrial use cases, will finally be discussed in the chapter 6.

Chapter 1

State of the art

During the last decades, the main driver behind the industrial robot deployment has been the desire to reduce or eliminate dull, dangerous, and dirty manual jobs while achieving higher quality and consistency in the manufacturing flow. First in the automotive industry and recently in the largest production plants and logistics facilities, robots became standard technology, a synonym of higher flexibility and better production. However, the growing demand for mass customization is rapidly and ineluctably substituting mass production, while the robotic manipulators are increasingly more common in small and medium-sized businesses too, to support the productivity. As discussed in [1], agile and flexible small and medium manufacturers are more resilient and growing rapidly than in the past, thanks to digital technology. This trend became sharper with Industry 4.0, which has dramatically transformed our world in recent years. Production plants took the form of Cyber Physical Systems (CPS), the key objective of which is to create a highly integrated, intelligent and automated manufacturing system that would enable the efficient use of data and technology to optimize productivity, increase efficiency, reduce waste, and drive innovation. This paradigm focuses on integrating advanced technologies such as artificial intelligence (AI), the Internet of Things (IoT), digital twins, robotics, automation, and big data analytics to create more flexible and responsive supply chain and production ecosystems. Products and services can be quickly customized with respect to the past to meet the changing needs and preferences of customers. By adopting smart technologies, manufacturers can remotely monitor and control the entire production process, reducing the need for human intervention, and ensuring that production continues uninterrupted. According to [2], for many producers, the new fully automated manufacturing system provides an opportunity to reduce costs while ensuring product differentiation, rather than adding value. However, in the digital economy, a winner-takes-all model leads to the emergence of technological monopolies and significant wealth inequality.

Despite the techno-economic vision of Industry 4.0 offering a great leap in optimizing the production processes and business efficiencies, the purely technological focus has completely overlooked the social impact. Therefore, it is not fit for purpose in the context of a climate crisis and planetary emergency, nor does it address deep social tensions. The authors of [3] stated that to ensure the well-being of workers, the development of the manufacturing industry should prioritize finding a balance between industrial goals and the needs of employees. This can be achieved only through the implementation of adaptive automation systems that involve and prioritize human input.

1.1 Human-centered industry 5.0

The human-centricity has gained momentum in recent research and, according to the authors of [4], represents an evolution of the current mindset, while the value of human worker knowledge and problem-solving skills increases significantly when effectively combined with automation solutions. The European Commission, following this trend, has formally discussed in [5] the need for a more balanced governance that considers all the aspects that would lead to more sustainable and responsible production. In the manuscript completed in December 2021 [6], the transformative vision for the European industry is identified and compared with the current Indsutry 4.0. The analysis acknowledges the current paradigm as technology-centered, leveraging digital connectivity and artificial intelligence for better efficiency, optimizing business models within the capital market dynamics, and minimizing the cost for maximum profit for stakeholders, thus neglecting the environmental, climate, and social impact. The vision of Industry 5.0, represented by the European Commission in [7] in Fig. 1.1, in fact, is centered on three interconnected values: human centricity, sustainability, and resilience. This means combining the competitiveness of the production system with sustainability and resilience, relying on the empowerment of workers through the use of digital technologies, and embracing a human-centered approach. Industry 5.0 has to become a post-capitalist model that ensures proper feedback loops between industrial transformation and a re-evaluation of capital, including natural and human capital flows. To achieve the desired level of agility, it is essential to develop a portfolio of research and innovation projects in addition to the necessary new policies.

Figure 1.1: Core values of Industry 5.0 [7]

According to [8], the necessary change in perspective from a technology-driven industry to a value-driven one will require a transformation of the industrial roles.

The use of technology in manufacturing must prioritize the needs and diversity of workers and society as a whole, ensuring physical and mental health, well-being, and fundamental rights such as autonomy, dignity, and privacy. It must be adaptable and fluently synergize with humans, exploiting strategies that involve operators in the design and deployment phases of smart industries. Technology alone cannot replace human wisdom and creativity, therefore, democratized factories should be able to promote and empower workers to a higher level of skill usage, to create meaningful connections and interactions by leveraging advanced technologies. This means that the main enabling technologies do not differ significantly from those previously considered, if not in their intended use. As reported in [8, 9, 10], they are mainly inherited from Industry 4.0 but are devoted to a smooth transition and a more socially and ecologically focused productive system. In fact, to contextualize them in a value-generating model, as represented in Fig. 1.2, an appropriate policy is required.

Industry 5.0 places humans back at the core of industrial production, with the help of tools such as collaborative robots and advanced sensors for human-machine communication. This not only provides customers with the products they desire but also offers employees occupations that are more meaningful than factory jobs have been in over a hundred years. Researchers in this field, as the authors of [11], agree with the fact that the incorporation of collaborative robotics in the manufacturing system is a significant socioeconomic change, if robots are used to help people with monotonous or dangerous tasks instead of replacing human labor. However, focusing on the present use of collaborative robotics, the work [12] reveals a gap between the potential and actual applications of cobots in real automation scenarios, highlighting barriers beyond typical challenges to the adoption of a new technology. The study highlights the need for improved cobot design and human-robot collaboration, necessitating further research and design efforts. Similar conclusions are drawn in [13], where the authors explore the integration of cobots into Industry 4.0 manufacturing, assessing the impacts on worker skills such as substitution, deskilling, reskilling, and upskilling. This paper emphasizes the need for technological empowerment in order to maximize the impact of cobots on workers' abilities. However, many companies are still in the early stages of cobot implementation and instead are concentrating on replacing workers. The findings demonstrate how each type of human-cobot interaction influences the skills of workers in various manufacturing activities.

The authors of [14] and [15] explored the interesting implications of implementing collaborative robots along with the skilled handcrafts technique to produce artworks. Utilizing industrial robot tools has the potential to create a gap between the craftsman and tangible contact with the material, as humans must always be kept away from the robot for safety purposes. Cobots, on the other hand, allow for a reduced barrier and closer human presence to the material, and with enough safeguards applied, could allow a human to touch the material while the robot is acting upon it, allowing the user to closely monitor the process to minimize errors, but also permit explorations for a better flow of work. The research result is a hybrid digital craft approach to collaborative robotic pattern making and handcrafting. The fabrication system reduced the amount of time and physical exertion in designing and cutting patterns from various materials. This demonstrates that collaborative robotic tools can

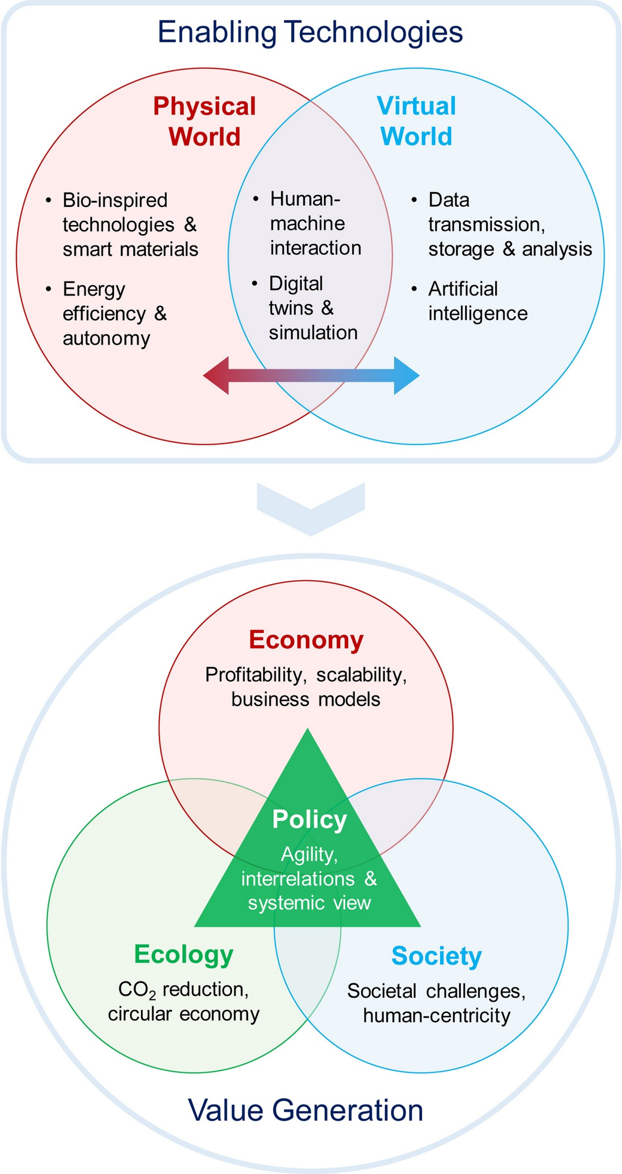


Figure 1.2: Goals and technological enablers of the Industry 5.0 model [8]

augment, rather than substitute, the abilities of existing artisan professions, aiding in the fortification of local industries and the encouragement of innovation. The necessity of enabling a perpetual interchange of inputs and feedbacks, investigating other augmented interactions between a human and a cobot, different professions, methods, and materials has been highlighted in [14] as essential for further investigating the benefits of cobots.

The authors of [11, 16, 17], recognizing the need for clear design criteria, highlighted the fundamental aspects related to operator safety, ergonomics, flexibility, and HRC for the design and implementation of a collaborative work cell, translating them into guidelines, requirements, and constraints. The conceptual HRC design presents some common aspect to be investigated in order to make the HRC effective and efficient:

- Possible product redesign for HRC;
- Definition of the robotic system: robot, end-effector, sensors, etc.;
- Definition of the safety system: hardware (safety fences, vision-based systems, etc.), and software control system (software limits and virtual fences);
- Definition of workstation elements: tools, equipment, etc.;
- Task allocation: human, robot, or human and robot;
- Design of the new layout considering several requirements and constraints (robot workspace, nearby workstations, AGV corridors and others).

The primary human factors that receive significant attention in collaborative applications are safety, physical ergonomics, and mental workload. Exploring these fields has resulted in improved production processes at safety levels, thereby increasing operator confidence in collaborative environments. Additionally, improved safety conditions allow the worker to enter the work area, hence directing and overseeing the improvement of automated processes. Although safety allows for the passive human presence in the working environment of machines and robots, suitable human-machine interfaces (HMIs) are required to actively exchange information between humans and machines. The development of user interfaces (UI) that are suitable for collaborative tasks and activities has been undertaken in many applications, including robot teaching pendant controllers, vision recognition systems [18], speech recognition systems [19], and other types of interface.

1.2 Cobots perception

Human-Robot Collaboration (HRC) seeks to complement traditional robotics by increasing human involvement in terms of shared time and space [20]. Through HRC, humans and robots can combine their respective strengths, provided that the machines involved are designed for both safety and interaction. The development of collaborative workstations requires close and secure interaction between humans and robots. To take advantage of this, much effort is devoted to research on human-aware robots, which can handle the unpredictability of human presence [21, 22, 23] and receive commands and guidance from them [24]. Different types of collaborative strategies are presented in [25].

- *Coexistence* is defined as the case in which the robot and the human operator share the work space, but not the process or the product on which they are working. This layout generally does not foresee direct interactions between humans and robots.
- A *synchronised* setup is the one in which the robot task starts after the worker has completed his operations or the other way around.
- *Cooperation* takes place when a human and a robot work in the same space and on the same products but perform different tasks.

- *Collaboration* is addressed as the completion of a shared task between humans and robots, where the actions of the former produce direct consequences of the latter, making the task more fluid, but also more complex, requiring additional sensors, calibrations, and programming.

The article [26] proposes a methodology to implement a modular control architecture based on the integration of recent advances in knowledge representation, task and motion planning, and human–system communication. The authors showed that the integration of these cutting-edge technologies lays the groundwork to push a paradigm shift in human-robot collaboration towards contextualized and user-centered production processes. To support this, the authors of [27] have suggested that programming a collaborative robot requires giving it the ability to recognize and understand its environment while performing activities that help achieve predetermined goals. However, conventional programming practices typically involve an offline procedure, limiting flexibility and failing to account for human presence in the robot's environment. Furthermore, these programs are challenging to modify during execution. The work [28] points out how important it is to consider the adaptive process in many environments where humans and robots physically interact. The results of the study also suggest that transitions between leader/follower modes influence performance, and subjects tend to prefer collaborative modes to those in which the robot assumes a fixed role.

To automate vision-based or contact-based processes, a variety of challenges must be addressed, including path planning, force and position control, and selection of process parameters. An important requirement for these processes is knowledge of the geometry of the workpiece and its position with respect to the robot, which is needed for path planning and process monitoring. Additionally, recognition of the specific characteristics of the items may be issued. With reference to additive manufacturing applications, for example, which require a high-precision trajectory, it is common to have prior information on the working object using computer-aided design CAD or technical drawings. In such cases, a solution based only on CAD information is required to generate the working trajectory as proposed in [29]. A relevant enhancement, described in [30, 31, 32, 33], consists of the possibility of visually identifying the shape and position of the workpiece, thus automatically defining the robot path based on the result of visual techniques. These data can be acquired through various means, such as 3D vision systems or tactile measurement systems.

1.2.1 Vision systems and haptic feedback

In order to automate the production processes that are still done by hand, it is essential to use flexible and adjustable tools that can be implemented quickly to accommodate the ongoing changes in the product. Among the suitable tools to enable such ability, vision systems have undergone a huge evolution in recent years, allowing one to automate several steps in production systems. In the robotic field in particular, the use of 3D vision systems has shown many capabilities, allowing the detection of characteristics of objects to be manipulated, such as color, shape, position, and dimensions, and therefore allowing the automation of applications in which the position and orientation of the manipulated objects must be identified [34].

Different approaches have been suggested to integrate CAD data with information from vision sensors. In [35], the welding trajectory is created offline from CAD and the data obtained from a 3D vision sensor is used to adjust it according to the current position and orientation of the object. A similar solution was proposed to automate the deburring application in [36]. In [37], a 3D vision system is used to construct the 3D model of an object, which is then used to generate the trajectory. Solutions that rely only on the data obtained from a vision system were also suggested. In [38], the welding path is created by detecting V-shaped grooves on the surface of the item to be processed using a 3D vision system. The system reached good performance when flat objects were considered and lower accuracy with non-flat objects. In [30], a robotic contactless glue application system was successfully implemented. This was achieved by creating the trajectory to be followed solely from the data obtained from the 3D camera. However, contactless glue deposition is less challenging than contact-based techniques because errors of a few millimeters are acceptable to achieve good glue deposition quality. It would be a huge improvement to be able to precisely recognize and locate the workpiece in the space and consequently provide a suitable trajectory and robot path, allowing at the same time to improve the process by collecting feedback evaluations from the operators. Moreover, eliminating the need for using CAD models, which are not always an exact copy of the physical product, would increase the error tollerance and overall production efficiency.

A major aspect of contact-based robotic processes is the control of the force and position. It is possible to identify two methods to apply active compliance: through the arm system and around the arm system. The through-arm system realizes force control by acting on the joints of the robot itself, while the around-arm system employs an active flange for force control while the robot follows a path [39]. The two most commonly used forms of force control are hybrid force-position control and impedance control[40, 41].

1.2.2 Advanced interaction interfaces

The development of various communication technologies is underway to enable real human-robot collaboration. Gesture recognition [42, 43], vocal commands [44], haptic controls [45] and the brain-computer interface [46] are some of the methods that can be used to give instructions to the robot, allowing operators to modify the robot's behavior during its operation or take control of the task [24, 47, 48]. These technologies can also be combined into a multimodal communication strategy, allowing more adaptive and unstructured programming of robots. The paper [24] discusses in detail the various techniques for human-robot interaction, proposing a metric to evaluate the performance and the type of information exchanged between humans and robots. The messages are classified as "Command" and "Guidance" messages, which can be achieved through different interfaces. Command messages are the simplest form of interaction, as they only require the robot to be given simple commands such as "next" or "stop", without any parameters needing to be exchanged. Guidance messages, on the other hand, involve the robot being given instructions on how to move, which requires a continuous flow of positional or force information. Therefore, a suitable design of the programmed task is necessary for command messages. In [27], the authors provide a detailed review of cobot programming, highlighting a significant gap between the solutions implemented in industry and those provided by researchers. The authors find that the use of cobots in collaborative industrial tasks has two main elements that are difficult to implement.

1. ensure that the operator can compose and alter dynamically the operations of a cobot in an intuitive way;
2. the awareness of the cobot of human presence to enable a more flexible and adaptive behavior.

The second aspect takes in account the awareness of the robot of the nearby human, which leads to the advantage of having an intelligent and experienced worker which can provide problem solving skills with no computational effort. Thus, we can affirm that the simplicity in programming a cobot is surely an advantage, but the online and offline interactions from the operator point of view must be considered, providing suitable hardware and software tools to support the collaboration, cooperation, synchronization, and coexistence, accordingly with the application.

It is essential to devise an efficient communication plan and user interface for the worker to collaborate with the cobot. Task-oriented programming, which is based on a hierarchical structure of fundamental abilities, can be employed to make cobot programming more straightforward and to allow the operator to directly interact with the robot by providing motion commands or input values to achieve the desired result.

The authors of [18] introduce a 'Meta-Collaborative Workstation' and a gesture-based robot program builder software, which provide a novel way of communication through the use of hand gestures. By combining gestures, the operator can create a personalized set of commands. The authors of [19] suggest a convenient and quick approach to program and communicate with cobots using voice commands. This method offers a hands-free experience, which is beneficial for workers who tend to have their hands occupied. The paper [49] discusses the use of the brain computer interface (BCI) to control a mobile robot on a construction site where an unmanned ground vehicle (UGV) is controlled by brain signals to delegate heavy load displacement to the mobile robot while the operator can decide where and when to guide the robot. The BCI device itself can be easily stored in the worker's protective helmet, allowing discrete placement of the electrodes. A BCI using steady-state visual evoked potentials (SSVEP) has been considered in [50] and [51] to drive a robotic arm that was able to move forward, backward, to the left, and to the right, and stop. High accuracy was achieved for the commands and zero class recognition was tested, which means that the robot stopped with high reliability if the subject did not watch the stimulation LEDs. To classify electroencephalography (EEG) data, Minimum Energy and Fast Fourier Transformation (FFT) with linear discriminant analysis (LDA) have been used. These results demonstrated that a SSVEP-based BCI could provide accurate and efficient high-level control of a robotic arm, showing the feasibility of a BCI-based control system for hand assistance. In [52] the authors prepared an overview of the use of BCI in the literature proposing a hybrid BCI based on electrooculography (EOG) and a visual tracking control model to guide an industrial robot.

The literature on EEG signal analysis, often used as a control interface, is abundant. Robotics applications have been developed in different contexts [53, 50]. Despite the potential of BCI technology to facilitate cobot collaboration in industrial settings, there is a dearth of discourse on how to effectively deploy it. There is no clear consensus on how to incorporate these devices into the collaborative layout of a manufacturing facility.

1.3 Programming methods

As defined in [27], the programming process of a cobot requires it to have the ability to understand its surrounding environment and perform actions that advance the system toward a planned goal. Traditionally, the programmer has only been involved in an off-line programming process. This typically results in programs that are inflexible, do not take into account the presence of humans in the robotic environment, and cannot be easily changed while running. The collaborative programming, instead, involves the operator in an on-line programming process allowing him to modify or affect a cobot program either explicitly or implicitly. Explicit participation refers to direct communication, whereas implicit interaction occurs when the cobot is able to extract information from its environment, adapting its policy accordingly. The policy can be learned from prior data, using advanced programming methods, or modeled manually by programmers. Therefore, it is evident that the cobots need to be equipped with intuitive interfaces to allow the operators to alter, create, and customize programs, either off-line or on-line.

Research concerning user-friendly programming environment and languages is deeply carried out in the literature, and many proposals have been made to lower the cognitive effort required to beginner level programmers who need to develop common robotic routines. In recent years, particular attention has been paid to visual programming approaches, which are becoming a standard programming method for collaborative robots. However, only a handful of studies have looked at how visual programming tools have been used in a specific domain such as robotics. This reflects on the key challenge of avoiding high-level expertise requirements to program a robust production-quality robot system. In [54], the authors highlight the advantages and challenges of a simplified collaborative robot programming method used in a real industrial application. The focus is on the identification of key features for a robust industrial program and how these characteristics may be implemented by any-level user. The review study [55], after analyzing various visual programming languages (VPLs), examined how each of the revised works performs in the classification, interaction style, target users, domain, and platform dimensions of the VPLs, considering the evaluation methods of each study. The conclusions of the work reported that VPLs make it easy for end users to visualize programming logic and eliminate the burden of handling syntactical errors.

The authors of [54] identified some of the major difficulties in creating robot programs, distinguishing between the creation and maintenance of the structure of the logical program and the creation of robot paths. A tool that makes it easy for a

novice user to create an application might not offer much support to an intermediate or advanced user for a more complex task, and vice versa. Similarly, some approaches can simplify the creation of a complex path, as the tools described in [56], but do not make it easy to create program logic. Although many easy robot programming approaches have incorporated more graphics than their text-based predecessors, adding graphics alone does not solve every ease-of-use issue. In the end, the design decisions made by any easy robot programming approach must be governed by a thorough user understanding of the system, the context, and the overall goals.
Following this idea, in [57] a visual task-level programming framework has been proposed to provide workflows for both experts, to create skills and provide their parameter interfaces, and for shop floor workers to use these skills to create executable robot tasks in an intuitive human–robot interface (HRI). In this system, users have the ability to modify tasks, delete or rearrange skills with drag and drop, and adjust parameter values without having to go through the entire wizard again. The professionals, in contrast, possess a variety of tools that enable the incorporation of any Human-Machine Interface (HMI) and the instruction of inexperienced users in the process of parameterization. Programmers are not limited to any particular coding language; yet, after they have written a skill, they must spend extra time to create a suitable parameter interface for the robot's eventual users.
It is interesting to note that in the evaluation of the framework proposed in [57], software has been reported to be more user-friendly by non-expert users than by programmers. Similar conclusions about usability can be extracted from [58], where the authors presented a VPL based on a flow chart for mobile manipulation tasks, named RoboFlow, which is claimed to be intuitive, designed to ensure robust low-level implementation and restrict high-level programming to avoid user errors. Although the proposed framework demonstrated that it can be quickly learned and easily used for common robotic tasks, with a low error rate and by any user, even expert programmers made some errors during the programming phase. In fact, even if they achieved good results in RoboFlow programming and found it intuitive, they were not in favor of using a visual programming language. Furthermore, during the interview, all six experts stated that they would rather write the same programs in a general-purpose programming language such as Python. The reasons behind such preferences were found in the fact that, in a VPL, certain abstractions are difficult to implement. In particular, once that the control flow merges at a node, no later transitions will be able to distinguish between the various possible execution paths that could have led to that node.

According to [59], the deployment of collaborative applications in the industrial domain exhibits increased complexity compared to traditional robot applications, and there is a notable research gap in terms of agility. This could be attributed to the general nature of agility, which might not have been thoroughly considered in collaborative applications. To overcome the need for complex programming, identified in the literature as one of the main problems preventing cobot diffusion into industrial environments, the paper in [60] proposes an approach to simplify the programming process, while still maintaining high flexibility through a pyramidal parametrized approach that takes advantage of cobot collaborative features.

Chapter 2

Safety standards

Robotic process automation (RPA) encompasses several critical phases, including accurate process identification and definition of objectives. Subsequently, robotic technology and the requisite devices are incorporated into a design that facilitates seamless integration and allows secure and compliant utilization. According to [61], one of the initial steps in the planning of a collaborative application is to determine the type of interaction required. To identify the type of collaborative operations, it is possible to rely on international guidelines, which are generally harmonized with law requirements on the safety level and help to understand the necessary safety systems to ensure compliance with the regulations.

The International Organization for Standardization (ISO) is a global association of national standards organizations. The task of creating international standards is usually assigned to ISO technical committees with the main objective of providing engineers with general structure and advice when constructing machinery, so that they can create machines that are safe for their designated purpose. The concept of machine safety takes into account the capacity of a machine to carry out its intended purposes throughout its useful life, while the risk has been sufficiently minimized. Although ISO standards are not mandatory, they are highly recommended to harmonize with law requirements and simplify technical support in the international market. The international standards form a series of norms structured in three main levels, as represented in Figure 2.1. Type-A standards establish fundamental concepts of basic safety, design principles, and general considerations applicable to machinery. Type-B standards are generic safety guidelines and address specific safety aspects or types of safeguards that can be used in a wide range of machinery. These are divided into two categories: B1 standards address particular safety issues, such as safety distances, surface temperature, and noise; B2 standards are concerned with safeguards, such as two-hand controls, interlocking devices, pressure-sensitive devices, and guards. Finally, Type C standards provide comprehensive safety requirements tailored to particular machines or groups of machines.

ISO 12100:2010 "Safety of machinery - General principles for design - Risk assessment and risk reduction" [62] is a Type-A standard. It is based on insights gained from machine design, use, incidents, accidents, and associated risks. This document provides guidance on how to identify, assess, and evaluate potential hazards and risks associated with a machine's life cycle. It emphasizes the need to eliminate

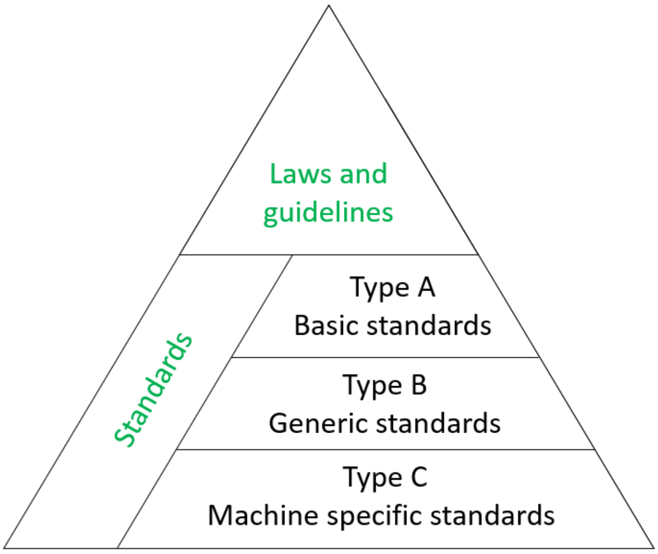


Figure 2.1: ISO standard levels

and reduce any risks that may be present. Furthermore, ISO 12100:2010 serves as the basis for developing type-B or type-C safety standards.
It is clear that ensuring the safe integration of robots, regardless of whether they are collaborative or not, is of paramount importance. It is essential to be aware that, although these standards are not compulsory, it is a mandatory requirement to carry out a risk assessment for all robotic applications. Risk assessment is a comprehensive and iterative procedure that typically involves multiple stakeholders, including manufacturers, suppliers, integrators, and users. It begins with the identification of any potential hazards related to the particular application or process being analyzed, followed by an estimation of its likelihood of occurrence and severity. Risk evaluations allows to decide whether a risk reduction is required or its frequency and extent are acceptable. Ultimately, the process focuses on the development of strategies to reduce risk. These include the inherently safe design, the only stage at which hazards can be eliminated, thus avoiding the need for additional protective measures, such as safeguarding or complementary protective measures. If the design is not enough to completely eliminate a danger, the right safeguarding and additional protective measures should be taken into consideration to reduce the risk to an acceptable level for the intended purpose. If the danger persists despite design decisions, safety measures, and the implementation of additional protective measures, the residual risks should be listed in the user instructions, thus informing the user of any remaining risks. The elements of risk analysis and assessment represented in Figure 2.2 are well defined in ISO 12100:2010.

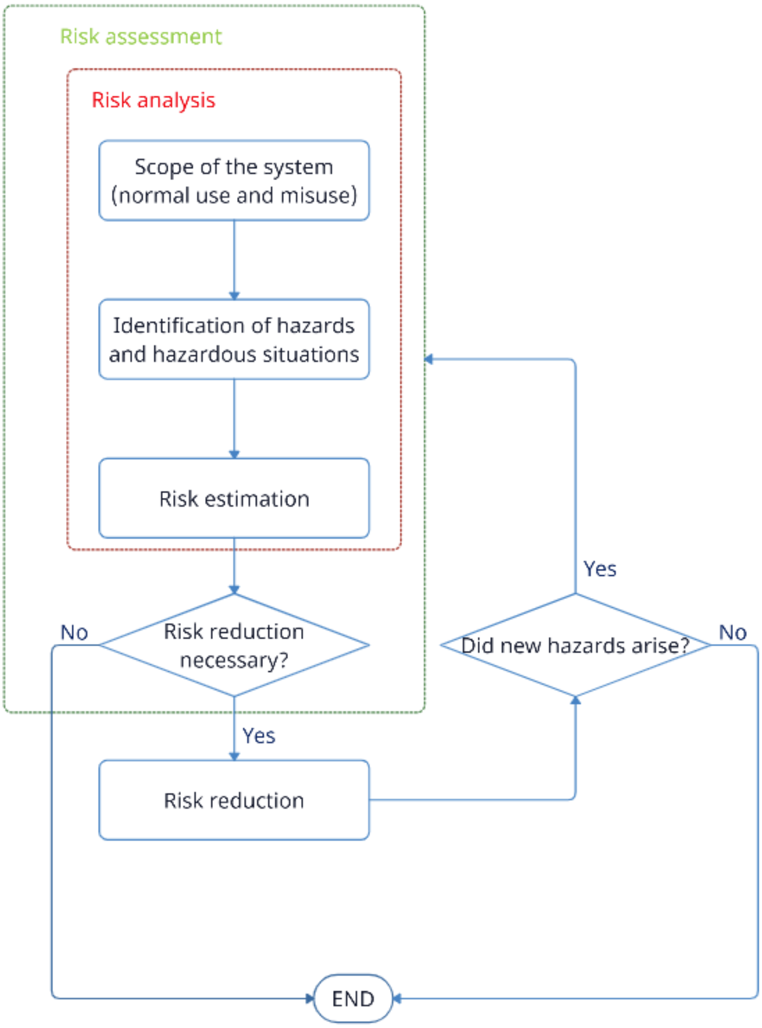


Figure 2.2: Risk assessment procedure

2.1 Safety requirements for industrial robots

To identify the risks that industrial robots and their systems may present, the ISO 10218 type-C standard was created. It is divided into two sections due to the varying risks associated with the use of industrial robots. The first part, ISO 10218-1:2011 ”Robots and robotic devices - Safety requirements for industrial robots - Part 1: Robots” [63] is related to the guidelines for the assurance of safety in the design and construction of the robot itself. This document outlines the necessary steps to ensure the safe design, implementation and use of industrial robots, highlighting the necessary precautions to be taken and providing information on the potential hazards associated with industrial manipulators. It also describes the steps to be taken to effectively eliminate or reduce these risks.

The level of risk posed by an automated process also depends on the features of the system and its complexity as a whole. In fact, the industrial robot is defined as ”automatically controlled, reprogrammable multipurpose manipulator, programmable in three or more axes, which can be either fixed in place or mobile for use in industrial automation applications" [63]. The robotic system besides the robot itself includes also the end-effector, identified as that device specifically designed for attachment to the mechanical interface to enable the robot to perform its tasks and interact with the environment, machinery, workpieces, or humans, and any other machine, equipment, device, external axis or sensors supporting the robot in performing its task. The potential risks associated with the use of robots depend on the type of robot, its purpose, and how it is installed, programmed, operated, and maintained. Since safety in the application of industrial robots is influenced by the design and integration of the particular robot system, ISO 10218-2:2011 "Robots and robotic devices - Safety requirements for industrial robots - Part 2: Robot systems and integration" [64] provides guidelines for personnel protection during integration, installation, functional testing, programming, operation, maintenance and repair of robots. In addition to the C-level regulations that may apply to other machines in a robotic cell, the robot system could be part of an integrated manufacturing system that is regulated by ISO 11161. This standard may also refer to other standards at levels B and C that are pertinent. However, additional considerations are necessary in the case of a collaborative operation. In this state, a robot system specifically designed for collaboration works in direct cooperation with a human within a defined collaborative workspace. It shall provide visual indication whenever it operates in collaborative mode and comply with one or more of the foreseen modalities.

2.2 Safety requirements for collaborative robots

Industrial robots are generally kept in cages for protection, away from human workers, and perform their tasks in isolation. Automatic operation, in fact, is meant to be the state of the robot in which it performs its programmed task in complete autonomy. To ensure safety, traditional robotic applications do not allow operators to enter the area where the robot is working while it is active. Consequently, many tasks that require human participation cannot be automated using common industrial robotic systems. Unlike their industrial counterparts, cobots are designed with human interaction in mind. The aim of collaborative robots is to bring together the repetitive capabilities of robots with the unique talents and aptitude of humans. People have an outstanding ability to tackle unstructured tasks, while robots demonstrate accuracy, strength, and stamina. To allow for this interaction, a comprehensive evaluation of potential hazards must be performed to analyze not only the robotic system itself, but also the environment in which it is located. ISO 10218 emphasizes that a robot, by itself, is not sufficient to guarantee secure collaborative operation; it is only a component of the entire robot system. Information for use shall contain a direction for implementing speed values and separation distances. Additional information is contained in the technical specification ISO/TS 15066:2016 "Robots and robotic devices - Collaborative robots" [65], which offers guidance for the use of collaborative robots, supplements the ISO 10218-1 and ISO 10218-2 safety standards, and provides instructions on the accepted operational capabilities of cobots.

It is essential to assess how often and for how long the operator is present in the collaborative workspace when the robotic system is running, taking into account the unpredictable nature of human behavior to reduce risks. Therefore, it is crucial to consider any additional human tasks performed within the collaborative workspace to ensure safety. A comprehensive evaluation of several key components is required to construct a collaborative robot system that is both secure and efficient. These factors include:

- Recognition of inherent robot hazards: recognition and mitigation of hazards intrinsic to robot design and operational characteristics.
- Recognition of system-related hazards: effective management of potential hazards that arise from the entire robot system.
- Recognition of application-specific hazards: understanding and addressing hazards that are unique to the intended applications of the robot.
- Workspace design: developing an ergonomic and conducive workspace that fosters seamless cooperation between humans and robots.
- Hazard Mitigation: identifying and eliminating potential hazards within the robot's operational environment to ensuring safety.
- Assessment of contact scenarios: a detailed assessment of scenarios that involve intentional or foreseeable contact between operators and the robotic system, along with measures to mitigate associated risks.
- Risk reduction: implementing measures and protocols to minimize any remaining risks associated with human-robot interactions.

The ISO/TS 15066 mainly concentrates on risk management and safety requirements for collaborative industrial robot systems. Potential hazards posed by machinery or equipment must be adequately addressed through risk assessment. The position of machinery and equipment should be done in a manner that does not create additional risks of body entrapment or crushing between the robot system and, for example, components of buildings, structures, utilities, other machines, and equipment. These risks must be eliminated or managed in a safe manner. However, the authors of [66] discuss the interpretation of the existing ISO/TS 15066 for collaborative robots, claiming the inconsistency of the terminology used in the case of transient and quasistatic contact events of parts of the robot system and the human operator. This can affect the risk assessment for dynamic and constrained contact scenarios. The final considerations of the work are related to the simplification of risk assessment procedures. Although considering any hazard is mandatory and reducing it by adhering to the ISO standards is a way to tackle many dangerous situations, the standards are only an element to be considered in the robotic cell design and are not sufficient to create an efficiently collaborative workspace. When to permit Human-Robot Interaction (HRI) and how to assign the task to each operator, human or robotic, is a complex area of research. Devices and interfaces are being explored to find the best recipe for seamless and intuitive HRC while ensuring safety and efficiency. The direction is set in expanding the intelligence and capabilities of robot systems to recognize and prevent hazards, thus focusing mainly on how not to harm people. However, the real outcome of the robotized collaborative station is the correct execution of the collaborative task, with improved characteristics due to the precision and strength of the robot. To fully valorize operator capabilities, the added value of human presence should be exploited by considering advanced strategies for robot utilization in a synergized manner. A skilled human operator can provide managerial abilities that the robot lacks and decision-making skills to the robotic cell. The standards today consider four primary collaborative modes, outlined in ISO/TS 15066 and ISO 10218, in the context of robot safety:

1. Safety-rated Monitored Stop (SRMS): when the robot system is operating in a shared workspace and the safety-monitored feature is activated, the robot's motion will be halted if a person enters the collaborative area.

2. Hand Guiding (HG): in this mode, the operator utilizes a hand-operated device to control the robot's movements. The robot system shall be equipped with an enabling guiding device.

3. Speed and Separation Monitoring (SSM): during robot motion, a crucial aspect is that the robot system never approaches the operator closer than a predefined protective separation distance. If this distance decreases below the set limit, the robot system is stopped. As the distance between the operator and the robot system decreases, the robot system slows down to protect the operator from potential harm. The maximum permissible speeds and the minimum protective separation distances in an application can be variable or constant.

4. Power and Force Limiting (PFL): in this mode, physical contact events between the robot system and the operator's body parts can occur intentionally or incidentally. It is imperative to conduct a comprehensive risk assessment and implement risk reduction measures to ensure that the limit values outlined in Annex A of ISO 15066 are not exceeded.

These modes are referred to as collaborative operations methods and are the reference for the deployment of robots in collaborative tasks. The robotic collaborative application, however, can include one or more collaborative and non-collaborative tasks. Where tasks are defined as portions of the robot sequence where both the robot and operator are within the same safeguarded space. Taking into account that SRMS and SSM are both designed to prevent the robot from moving when a person is near it, and HG and PFL are promising methods for enabling close physical interaction, a collaborative robotic cell, which halts the robot's motion when a human is nearby, can include a traditional non-collaborative robot. As a consequence, mode transition management is vital to ensure the safe switch between non-collaborative and collaborative operations of the robot. To guarantee safety, the stopping functions should be meticulously designed to enable operators to swiftly stop the robot's motion through a single action or exit the collaborative workspace unobstructed. These functions can include the activation of emergency stop mechanisms or manual robot shutdown procedures.

This approach is also effective in traditional industrial cells, where human presence is not allowed. The employees responsible for the execution of the process were promoted to process managers and supervisors, and the human role was shifted to programming, maintaining, and overseeing robots. The design of collaborative robots allows for close physical interaction, recognizing them as machines that can work together with humans in craft and artisan production. To enable people to take control of automated processes, in addition to technological advancement and increased safety, innovative strategies must be implemented to effectively facilitate HRI. The increased complexity of these environments requires operations that are adaptable and that need monitoring and safety verification while the robot process is being executed. The use of digital twin software tools to conduct virtual digital risk and safety assessments has the potential to significantly improve the interaction between humans and robots in collaborative robotic cells. The worker, in any case, must have the capability to control the machine's behavior in the desired manner, and this control must be intrinsically safe, under any circumstances. Many production processes, such as surface finishing, assembly, and glue deposition, require experienced personnel to perform the tasks by hand due to their complexity and difficulty in automation. Adequate communication means are required to collect human input and merge them with environmental data acquired by using a wide variety of sensors. These must be implemented alongside the robot to complete the automated system and guarantee the most efficient and successful completion of the task.

Chapter 3

Technological assets

In this chapter, the hardware and software tools used to develop the proposed solutions and test the developed industrial cells are introduced. The main laboratory research studies were carried out using the collaborative TM5-700 manipulator[67] described in the following subsection 3.1.1. To perform laboratory tests, feasibility studies, and the experimental campaign, in addition to the robot arm that is equipped with a built-in vision system, a series of sensors and end effectors were deployed. In particular, a six-axis load cell, described in Section 3.3.1, is used for force / torque measurement and hand guiding applications, while an external 3D vision camera, presented in Section 3.3.2, is utilized to expand the possibilities of vision-based applications. In addition, to test the possibility of using the brain computer interface in a collaborative human-robot task, a headset capable of collecting brain signals and transmitting them to a computer is used. The device set used to collect and transmit biosignals is described in Section 3.3.3.
Commercial OnRobot grippers, specially designed 3D printed tools, and extruders were used as end effectors to test the proposed layouts. These devices are reported in the section 3.2

To evaluate the effectiveness of the proposed paradigms in a commercial setting, a collaborative robotic cell was developed, tested, and implemented in partnership with a local business. To meet the needs of the industrial application being considered, a TM12 manipulator was used, which allows a large working area. The latter is described in Section 3.1.2. The task and workpieces required the creation of specific end effectors, which are fitted with all the necessary sensors. This is discussed in Section 3.2.

TMflow, discussed in Section 3.4.1, is the main software used to program the basic logic and movements of the TM robot. Additionally, open source libraries or programs are employed to provide signal analysis, simulations, or computations.

3.1 Collaborative robots

3.1.1 Techman TM5-700

The Techman Robot TM5-700 [67], shown in Figure 3.1, is a commercially available six-axis collaborative robot. TM Robot is made up of the robot arm and control box (including a robot stick). The TM5-700 has a maximum reach range of 700 mm, as shown in Figure 3.2. The payload capacity of the robot arm is contingent on its center-of-gravity offset, which is the distance from the center of the tool flange to the center-of-gravity of the payload. Its maximum payload is 6 kg for poses closer to the robot's center of gravity and gradually decreases as the distance from the center of gravity of the poses increases by more than 150 millimeters, as shown in Figure 3.3. The robot is equipped with an integrated vision system that enables it to recognize different items, perform self-adjustment, and visual tasks. The user-friendly interface and the hand-guided teaching mode make it easy for operators to operate the cobot. TMFlow is the dedicated programming software that has a simplified block-based programming language.

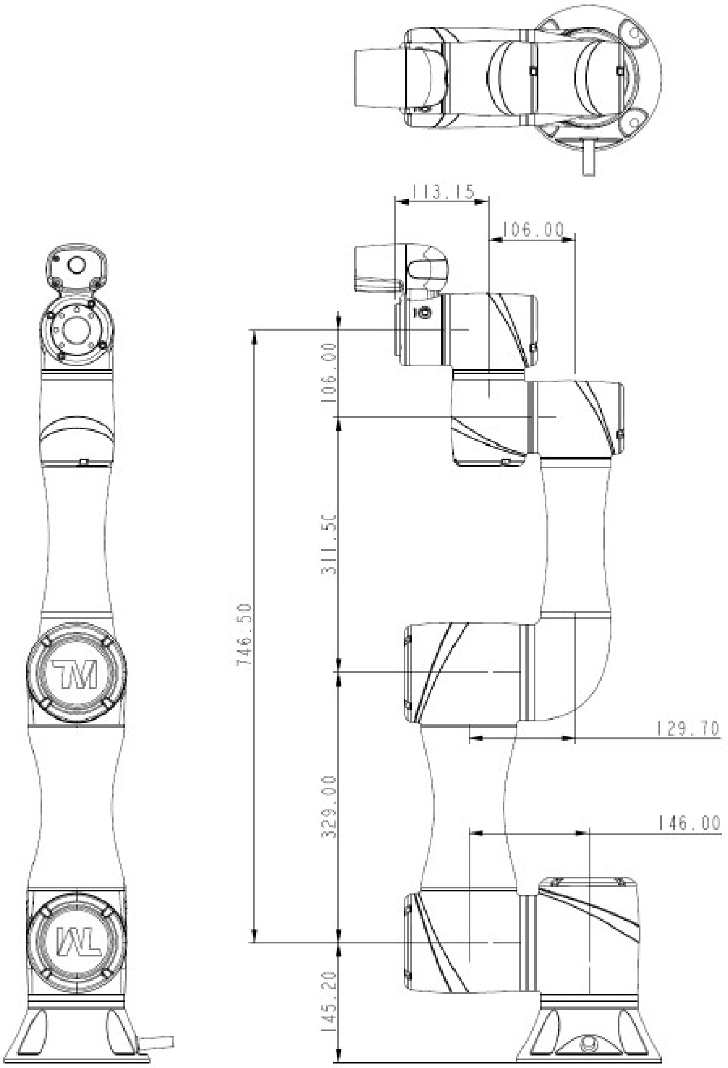


Figure 3.1: Schematic representation of the TM5 - 700 collaborative robot [67]

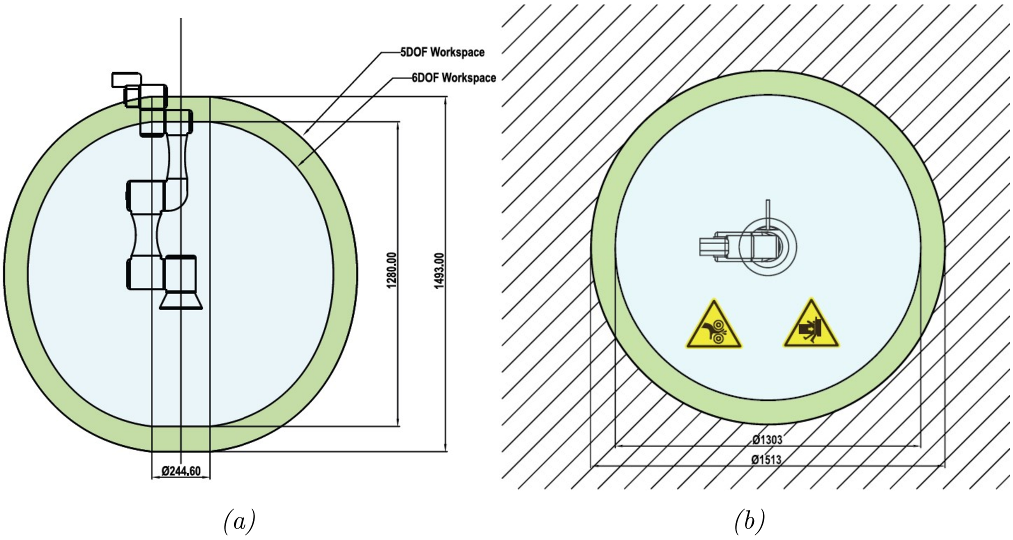


Figure 3.2: TM5-700 movement range diagrams a) Side view b) Top view [67]

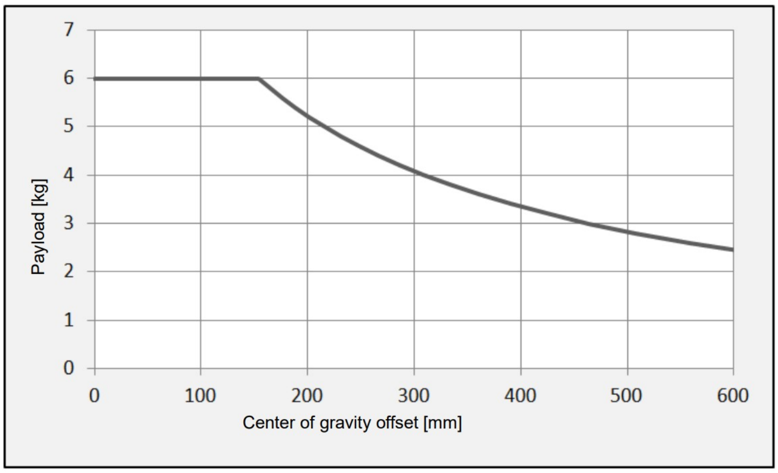


Figure 3.3: TM5-700 payload diagram [67]

3.1.2 Techman TM12

The TM12 collaborative robot [68] has similar characteristics to TM5-700 except for the dimensions and payload. The schematic representation of the TM12 cobot and its dimensions are reported in Figure 3.4. Figure 3.5 shows the working spherical range (radius) from the base, that is, 1300 mm for the TM12 series.

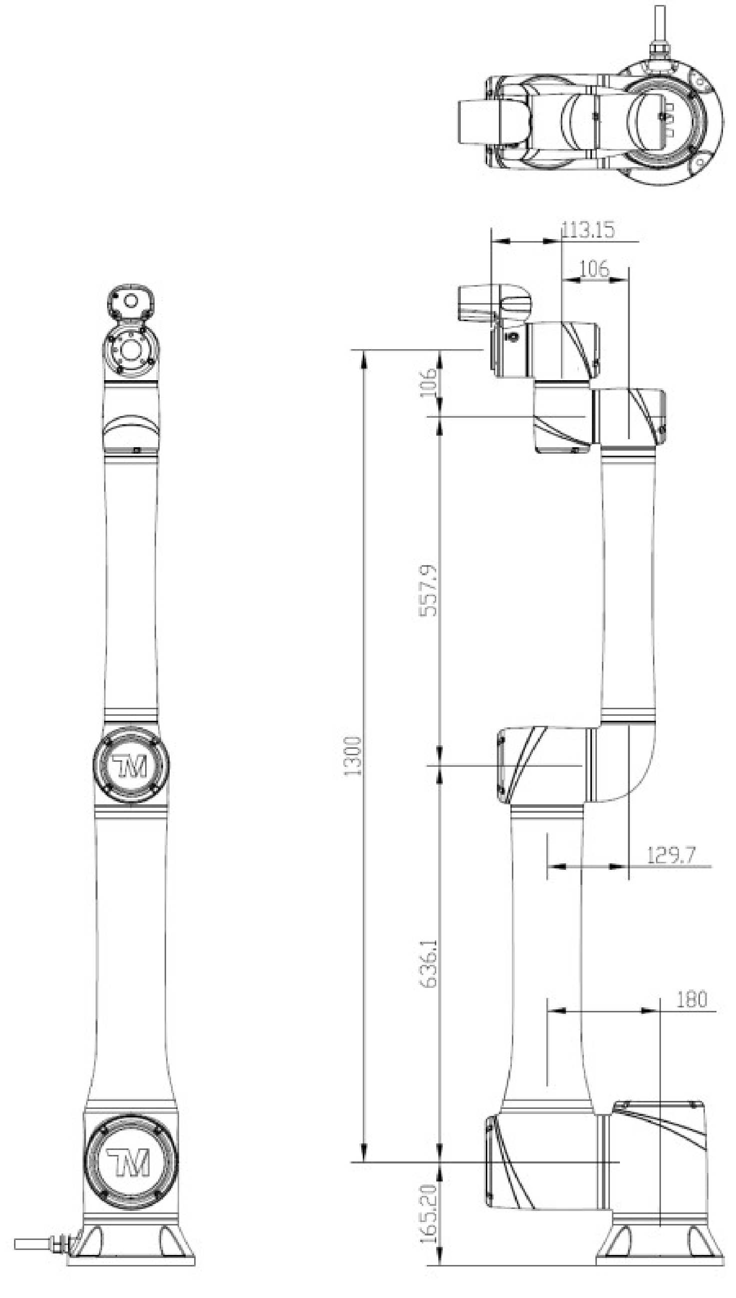
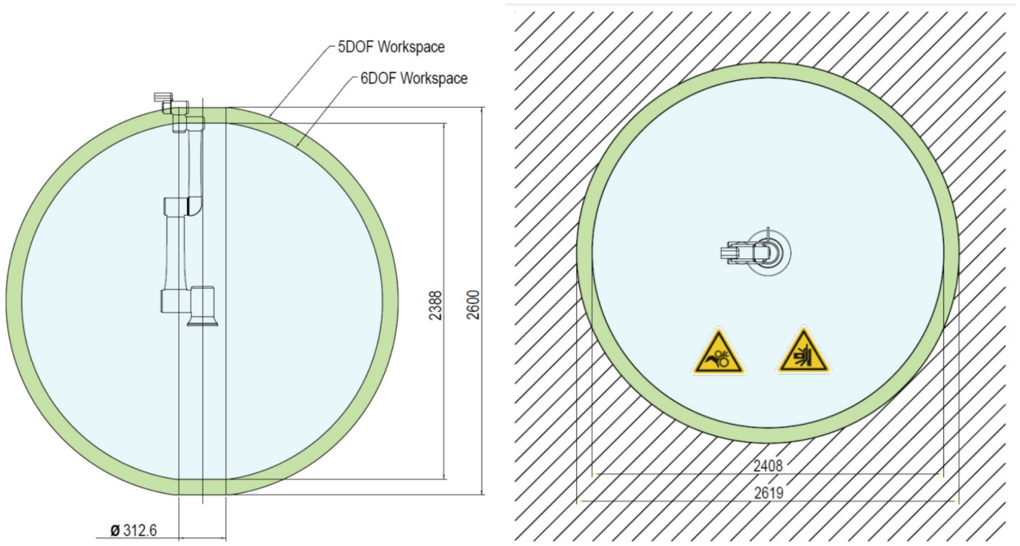


Figure 3.4: Schematic representation of the TM12 collaborative robot [68]



(a) (b)

Figure 3.5: TM12 movement range diagrams a) Side view b) Top view [68]

The maximum payload that the robot arm can carry is with respect to the offset of its center of gravity. The maximum weight that the robot can carry is 12 kg when the position is close to its center of gravity. As the distance from the center of gravity increases by more than 100 mm, the payload decreases, as illustrated in Figure 3.6.

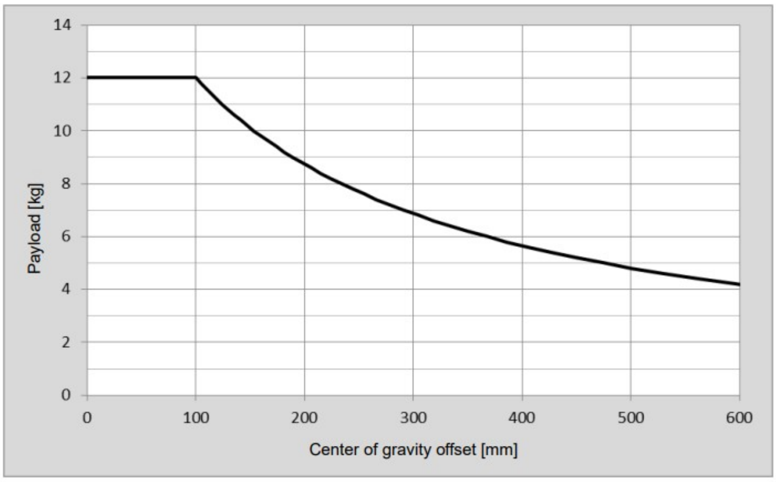


Figure 3.6: TM12 payload diagram [68]

3.2 End-effectors

3.2.1 Commercial gripper

On robot finger gripper

One of the grippers used to test collaborative laboratory applications is the OnRobot 2FG7 finger gripper [69]. It is a complete plug-and-play electric parallel gripper that requires no custom engineering for installation, programming, or maintenance. The 2FG7 is easily redeployed on any major collaborative or light industrial robot, making it ideal for low-volume high-mix production for many different applications. This gripper is perfect for use in applications with limited maneuver space. Boasting a maximum payload of 11 kg, an external grip range of up to 73 mm, and a gripping force of 20 N to 140 N. It can easily handle heavy or bulky payloads. The electric 2FG7 can be easily programmed for precise force, speed, and stroke control settings through an intuitive software interface. Intelligent feedback, such as grip detection and lost grip detection, improves overall accuracy.

Figure 3.7: OnRobot 2FG7 electric finger gripper [69]

In order to wire the system correctly, three cables must be connected. The first is the tool data cable, which links the tool(s) to the compute box. The second is the ethernet communication cable, which connects the robot controller to the compute box. Lastly, the power supply of the compute box (CB) must be connected. The schematic representation of the wiring is shown in Figure 3.8.

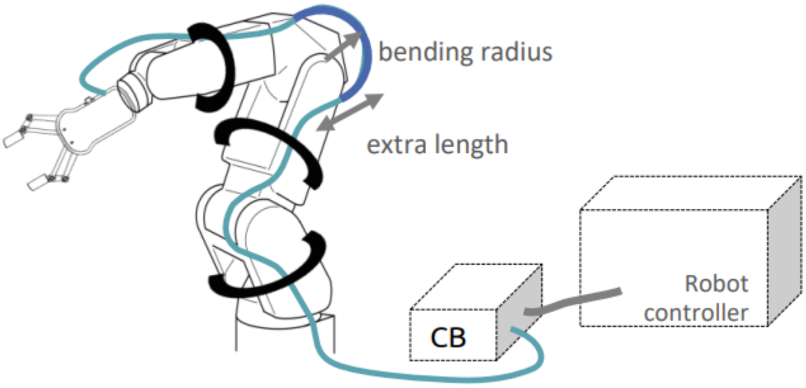


Figure 3.8: OnRobot 2FG7 electric finger gripper wiring [70]. The gripper is connected to the compute box (CB), which works as an interface to the robot controller.

3.2.2 Designed end-effector

Given the versatility of robotic systems, it is often necessary to create grippers that meet the demands of the process or product being handled. In the applications discussed in this work, especially in industrial settings, the potential to optimize the end effectors for the particular case was taken advantage of. The following subsections 3.2.2, 3.2.2, and 3.2.2 detail the designed end effectors.

Glue deposition system

One of the contact-based robotic applications developed is a glue deposition process in footwear manufacturing. In this application, a Fused Deposition Modeling (FDM) extruder is employed as the glue application system. The extruder is similar to 3D printing extruders and consists of a stepper motor that controls the filament feeding to a hot end, at the desired flow rate. The adhesive material is provided in the form of a filament with a diameter equal to $3mm$. The extruder motor is connected to a speed reducer (with a ratio of 1/3) which propels the filament directly to the hot end at a rate of approximately $2mm/s$. The hot end is capable of ensuring the temperatures necessary for the melting of the glue, between 285 and 295 ° C. The nozzle has a diameter of 0.8 millimeters. The control of the extrusion system is achieved by an Arduino Mega based board combined with a shield featuring as stepper driver (DRV8825 up to 1/32 microstepping), with a MOSFET transistor commanding a heat cartridge (12 V, 40W) and with a temperature sensor (PT1000) placed in proximity of the extrusion nozzle. The Arduino microcontroller and the robot controller communicate with each other through TCP/IP to exchange process parameters, coordinating the flow of adhesive material and the linear speed of the robot's movement.

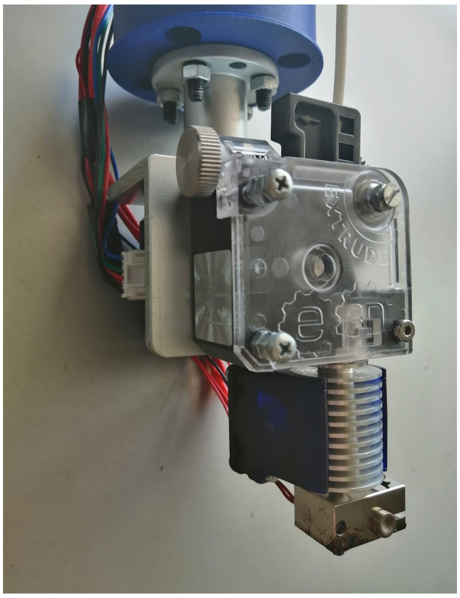


Figure 3.9: Glue deposition system

Vacuum grippers

In the industrial pick-and-place application developed, two distinct vacuum grippers were deployed to accommodate the shapes of two alternative items that the robot is meant to move. The gripper depicted in Fig.3.10a is used for the first part of the task. It has four vacuum cups and four 3D printed components, mounted in order to properly match the surface of the object to be picked up. Each of the two vacuum pumps is connected to a pair of vacuum cups and each pump is equipped with an electronic vacuum switch to verify suction. A sonic depth sensor is mounted on the gripper to identify the distance of any surface of an object in the range of 80-800 mm.
The other type of objects to be picked by the robot in a subtask of the same application have different dimensions, shape, and stiffness. To meet the features of the second type of item, another gripper was designed that features different numbers, configurations, and sizes of mounted cups, with the vacuum cups evenly distributed along a row. This alternative gripper is represented in Fig.3.10b.

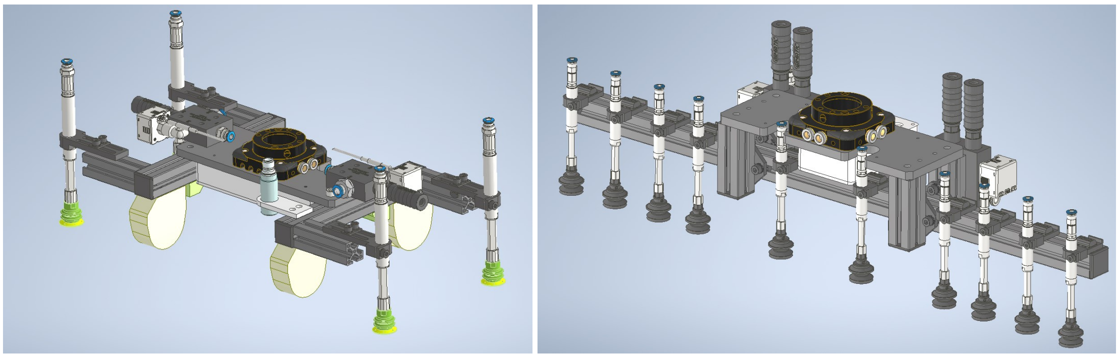

(a) (b)

Figure 3.10: The grippers developed for the application: a) the gripper for the trays displacement has 4 vaccum cups mounted in order to couple correctly with the surface to be picked; b) the gripper for the capsules pick and place task mounts 10 vaccum cups evenly distributed along the row of the stacked capsules.

To be able to automatically exchange the grippers in a changing station, both grippers in Figure 3.10 are equipped with the same connection flange. This is part of the Gimatic-commercialized KIT-TM-EQC20 [71] that is represented in the figure 3.11. The kit is an ”Electric Quick Tool Changer” that allows the end-of-arm tool to be easily replaced on the robot. It is specially designed for the entire range of TM cobots and consists of two parts: one permanently attached to the robot (EQC20TM-A) and one permanently attached to the tool (EQC20-B). By controlling the appropriate digital output, the two parts can be coupled or uncoupled for quick and easy tool changes. The entire system is a plug-and-play device that includes all the components needed to supply electrical and pneumatic power to the tool.

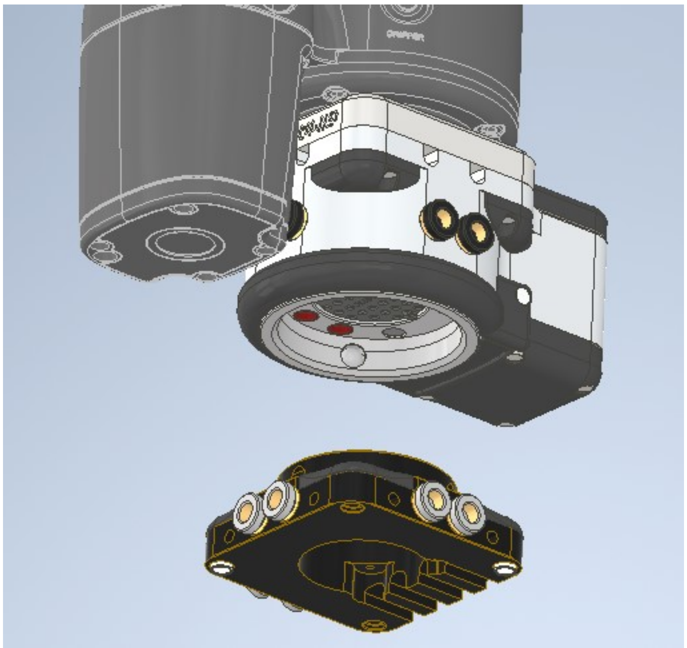


Figure 3.11: KIT-TM-EQC20 Electric Quick Tool Changer. It consists of two parts: one permanently attached to the robot (EQC20TM-A) and one permanently attached to the tool (EQC20-B).

Tray transporting gripper

A custom-made end effector has been created to facilitate the transportation of trays in an industrial setting. Figure 3.12 shows the tray gripped by the end effector attached to the robot flange. The purpose and further details of the application will be discussed in Section 5.2.

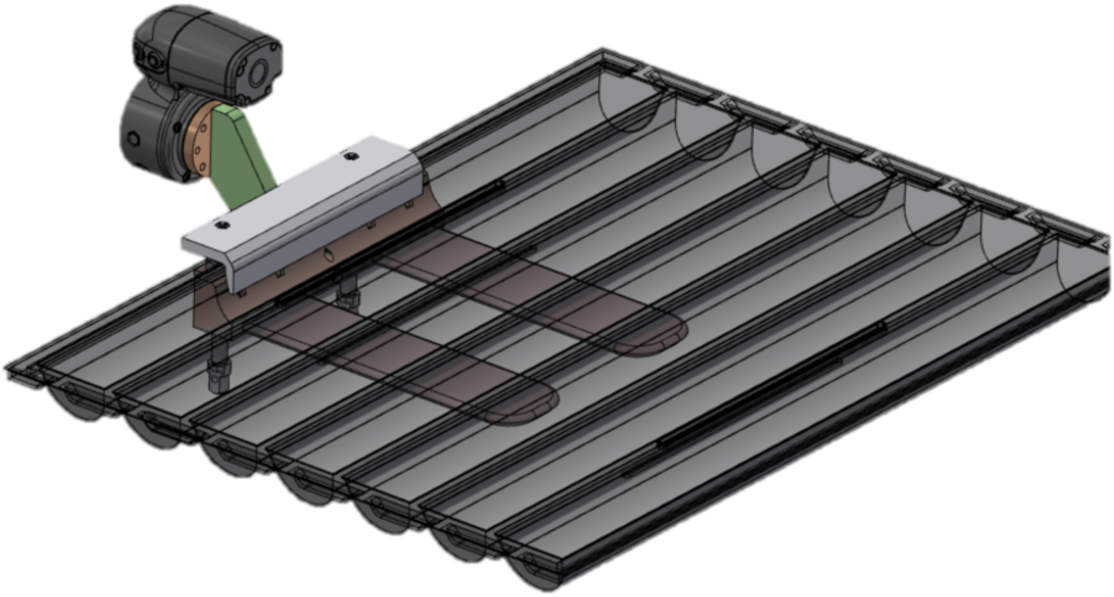


Figure 3.12: Gripper designed to carry the trays

Tray support has been foreseen to ensure proper insertion into their allocated positions, as they must maintain their straight shape and avoid any bending. A proximity sensor is mounted on the gripper to check the actual presence of the item during task execution. The end effector is represented in Fig. 3.13. The only moving part is the one that grips the tray and is moved by two pneumatic cylinders. The interface flange has been designed to be mounted on the standard TM collaborative robot flange.

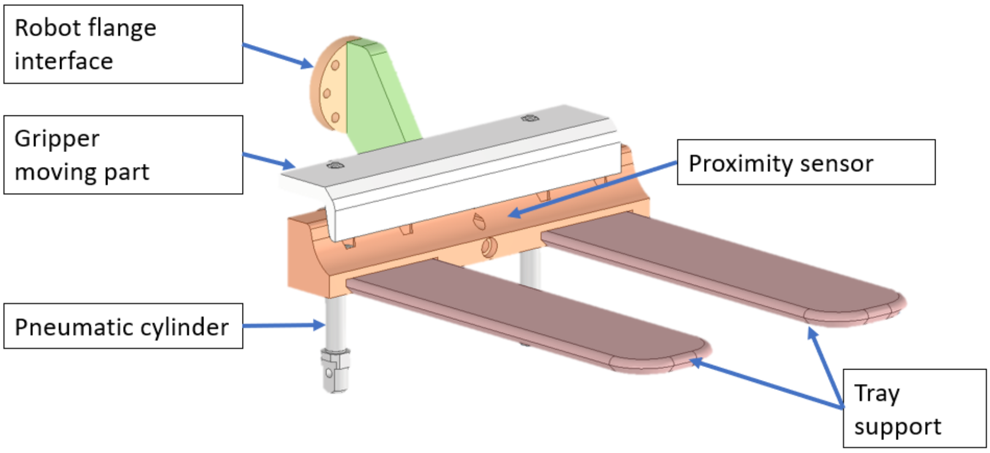


Figure 3.13: Gripper designed to carry the trays

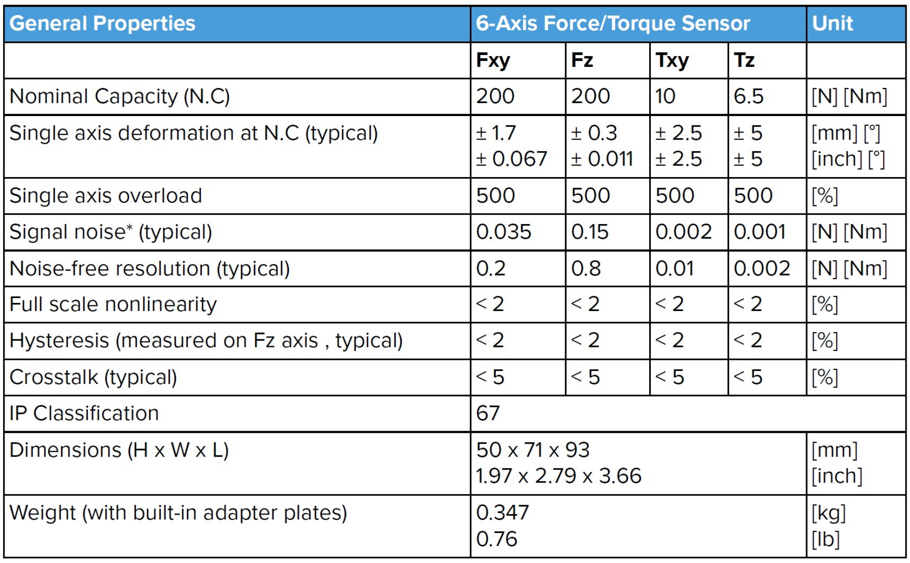
3.3 Robotic additional sensors

3.3.1 Force / torque sensor

The OnRobot Hex-E force / torque sensor is shown in Figure 3.14. This product is available for purchase from the OnRobot company. The sensor facilitates the use of robotic manipulators in applications that require a high degree of sensitivity and precision, such as insertion and assembly operations. Its nominal capacity is 200 N and it has a noise-free resolution of 0.8 N.

Figure 3.14: Hex-E force/torque sensor [70]

The axial direction has a capacity of 200 N, a signal noise of 0.15 N, and a noise-free resolution of 0.8 N. The technical specifications of the load cell are reported in figure 3.15, while the dimensions are reported in figure 3.16.



General Properties	6-Axis Force/Torque Sensor				Unit
	Fxy	Fz	Txy	Tz	
Nominal Capacity (N.C)	200	200	10	6.5	[N] [Nm]
Single axis deformation at N.C (typical)	± 1.7 ± 0.067	± 0.3 ± 0.011	± 2.5 ± 2.5	± 5 ± 5	[mm] [°] [inch] [°]
Single axis overload	500	500	500	500	[%]
Signal noise* (typical)	0.035	0.15	0.002	0.001	[N] [Nm]
Noise-free resolution (typical)	0.2	0.8	0.01	0.002	[N] [Nm]
Full scale nonlinearity	< 2	< 2	< 2	< 2	[%]
Hysteresis (measured on Fz axis , typical)	< 2	< 2	< 2	< 2	[%]
Crosstalk (typical)	< 5	< 5	< 5	< 5	[%]
IP Classification	67				
Dimensions (H x W x L)	50 x 71 x 93 1.97 x 2.79 x 3.66				[mm] [inch]
Weight (with built-in adapter plates)	0.347 0.76				[kg] [lb]

Figure 3.15: HEX-E technical specifications

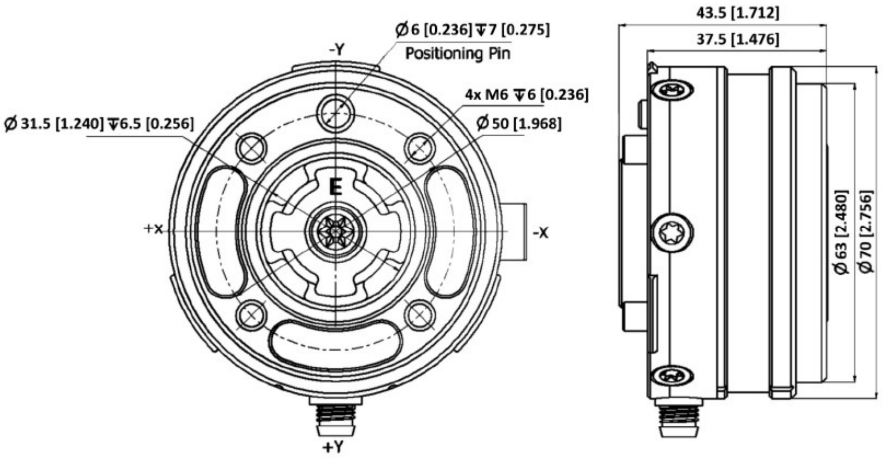


Figure 3.16: HEX-E force/torque sensor dimensions

The sensor is connected to a dedicated computer box (CB) that is connected to the robot controller through high-speed EtherCAT communication. The wiring is similar to that of the 2FG7 gripper described previously in Figure 3.8.

3.3.2 Vision system

Robots that handle, interact with, or come into contact with objects in a highly unstructured environment, such as a collaborative robotic cell where humans are present, can benefit from the use of vision systems to improve their interaction with the environment. A 3D vision system provides an advantage by providing depth information in addition to color data. This helps to plan the trajectory that a robotic tool must follow to complete a contact-based robotic operation. The 3D stereo cameras are based on the stereo depth principle to calculate the depth of each pixel in the camera's field of view. This method involves two vision sensors that capture two images from two different perspectives. By understanding the position of the two cameras relative to each other and comparing the positions of the pixels in the two pictures, it is possible to calculate the distance of all the pixels.

The D435 stereo depth camera [72] is a general purpose 3D camera suitable for robotics applications. This device has an 85 °field of view and can capture color and depth images with a maximum resolution of 1280 x 720 megapixels. The resolution is inversely proportional to the minimum distance between the camera and the workpiece. At maximum resolution, the minimum distance is about 280 millimeters. The frame rate is up to 30 frames per second (fps) for higher resolution configuration. The D435 camera is shown in Figure 3.17.

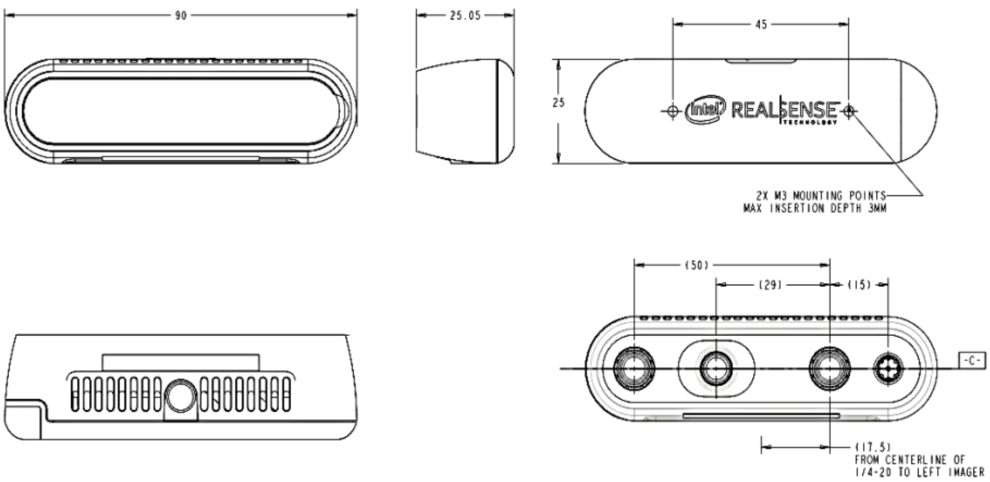
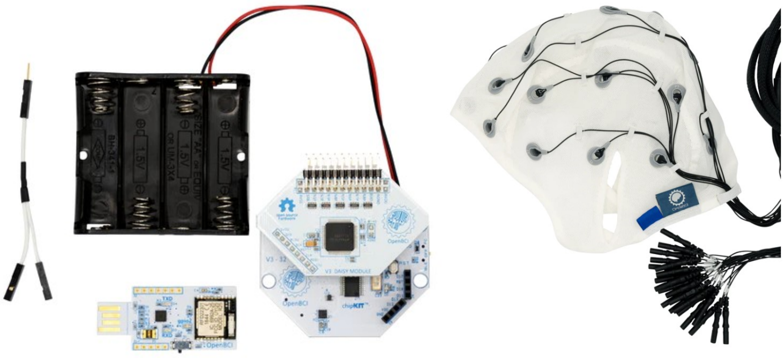


Figure 3.17: Intel RealSense D435 3D camera [72]

3.3.3 Brain computer interface

The OpenBCI Cyton board integrated with the OpenBCI Daisy module allows the sampling of up to 16 channels of brain activity (EEG). The Cyton-Daisy biosensing board is equipped with a microcontroller, which provides it with local memory and fast processing capabilities. The system communicates wirelessly to a computer through the OpenBCI USB dongle using RFDuino radio modules. The Cyton-Daisy board samples data at 125 Hz on each of its 16 channels. OpenBCI boards have a growing list of data output formats, making them compatible with an expanding collection of existing biofeedback applications and tools. The EEG Electrode Cap is linked to the Cyton-Daisy board to provide EEG bio-potential readings when utilizing wet electrodes. The headset, the board, and the usb dongle are shown in Figure 3.18.



(a) *(b)*

Figure 3.18: Brain computer interface headset a) OpenBCI Cyton-Daisy biosensing board and OpenBCI USB dongle b) OpenBCI EEG Electrode Cap

The OpenBCI Graphical User Interface (GUI) is the software tool used for viewing, recording, and streaming data from OpenBCI boards. It can be used to show data in real-time, replay it, save it to the computer in ".txt" format, and stream it live to third-party programs such as MATLAB.

3.4 Software tools

The proposed robotic applications are developed mainly in the proprietary TMflow robot environment, described in 3.4.1, in conjunction with the Python programming language to take advantage of several open source libraries to handle sensor signal analysis. Autodesk Inventor, a computer-aided design application, is used for designing the components of the custom grippers and the 3D printed parts. RoboDK simulator is used to design the robotic cell prior to its development. Additionally, the robot's reachability workspace and singularities are tested in the proposed industrial application by simulating the robotic cell.

3.4.1 TMflow programming software

Block-based programming, as described in [73], is a visual programming language that allows users to construct programs by connecting blocks that symbolize commands. This kind of programming tool can be utilized as a graphical user interface for programming collaborative robots, as it is easy to learn and use, even by those without any prior programming experience. Task design consists of the use of a graphical interface in which every robot action is represented in the form of a block. To program an application, the user has to drag and drop the blocks in the desired order.

TMflow [74] is a programming environment created to program and manage collaborative TM robots. This software has a user-friendly graphical interface that makes it easy to create, modify, and run robot programs. TMflow employs a flow-based programming paradigm with drag-and-drop nodes, making it accessible to users with varying levels of programming experience. The software offers extensive motion control capabilities, supporting both joint-based and Cartesian-based modes. TMflow also enables seamless sensor integration, allowing robots to interact with their surroundings based on sensory feedback. It includes error handling and recovery features and maintains a user-centric design with intuitive icons and tooltips. Although the specific functionalities may vary with software versions and robot models, TMflow provides a comprehensive solution for collaborative robot programming and control.

TMflow programming environment is shown in Figure 3.19. The blocks on the left side represent the different functions and can be dragged, dropped, adjusted, and linked together to create the robot's task. Programs with multiple threads, subtasks, and other capabilities can be used to create an efficient design for robotic processes and communication instances.

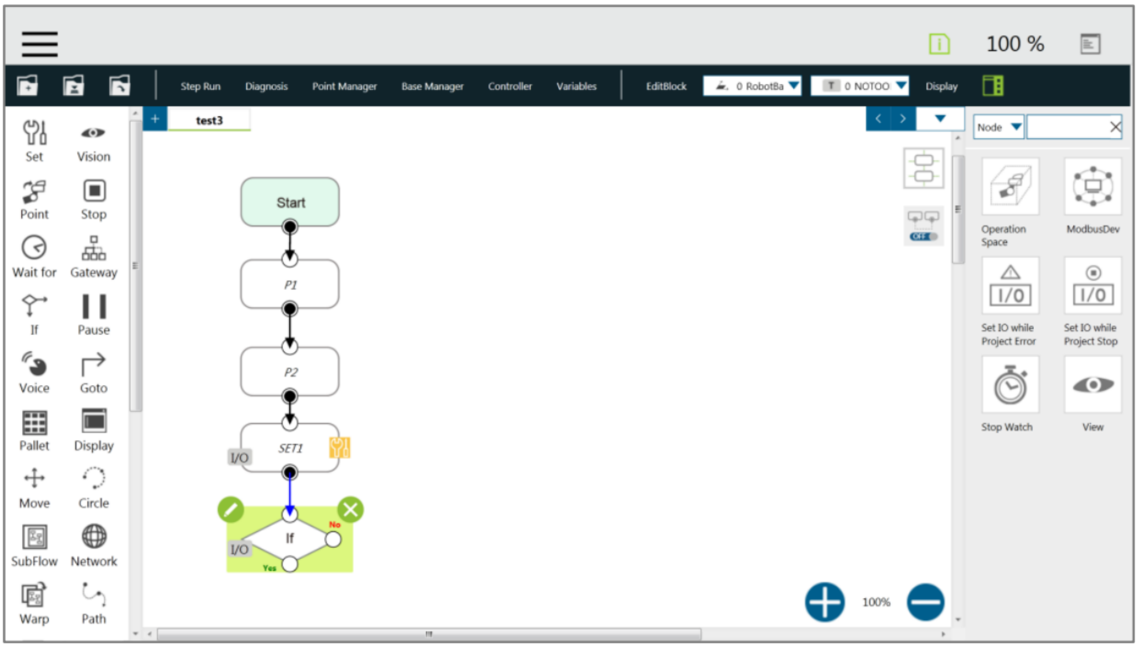
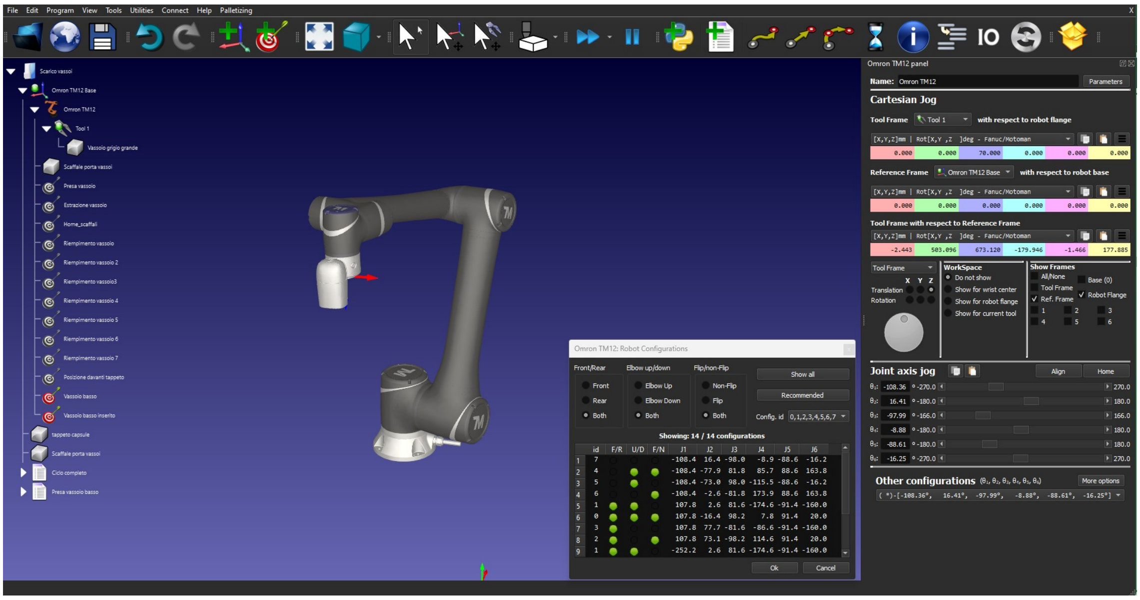


Figure 3.19: TMflow programming environment [74]

3.4.2 RoboDK

For industrial application design and development, the simulation and offline programming software speeds up a lot the cell design and allows for quick sizing process, movement testing, and studying multiple scenarios of a robot work cell before setting up the production cell. Mistakes commonly made in the design of a work cell can be predicted in time and avoided by optimizing the space usage and the characteristics of the robot's movement.

RoboDK is a versatile offline programming simulator for industrial robotics. It is free to try and is free for educational purposes. The software offers an extensive library of robot models and additional linear axes. It is possible to load a 3D model of a self-designed tool, convert it to a robot tool, and manually enter the tool coordinates (TCP) as in the robot controller. Additional items can also be located in the conceptualized robotic cell and create a complete digital twin.
All robots, tools, and objects have a relative reference frame and an absolute reference with respect to the cell. It is possible to quickly verify the proof-of-concept by simulating the movements in the robot coordinate system and also manually enter the coordinates as in the real robot controller.
Creating a robot path with the RoboDK user interface is a straightforward process. It automatically generates paths that are free of errors, while avoiding singularities, axis limits, and collisions. In Figure 3.20 an example of the RoboDK user interface is shown. On the left side is shown the dependency tree of all items loaded into the digital cell, including the robot, the target poses, additional reference frames, and the programs created. On the right-hand side are represented the robot panel and the configuration panel.



Omron TM12: Robot Configurations

Front/Rear: ○ Front ○ Rear ● Both

Elbow up/down: ○ Elbow Up ○ Elbow Down ● Both

Flip/non-Flip: ○ Non-Flip ○ Flip ● Both

Config. id: 0,1,2,3,4,5,6,7

Showing: 14 / 14 configurations

	id	F/R	U/D	F/N	J1	J2	J3	J4	J5	J6
1	7	○	○	○	-108.4	16.4	-98.0	-8.9	-88.6	-16.2
2	4	○	●	●	-108.4	-77.9	81.8	85.7	88.6	163.8
3	5	○	●	○	-108.4	-73.0	98.0	-115.5	-88.6	-16.2
4	6	○	○	●	-108.4	-2.6	-81.8	173.9	88.6	163.8
5	1	●	●	○	107.8	2.6	81.6	-174.6	-91.4	-160.0
6	0	●	●	●	107.8	-16.4	98.2	7.8	91.4	20.0
7	3	●	○	○	107.8	77.7	-81.6	-86.6	-91.4	-160.0
8	2	●	○	●	107.8	73.1	-98.2	114.6	91.4	20.0
9	1	●	●	○	-252.2	2.6	81.6	-174.6	-91.4	-160.0

Name: Omron TM12

Tool Frame Tool 1 **with respect to robot flange**

`[X,Y,Z]mm | Rot[X,Y ,Z ]deg - Fanuc/Motoman`

0.000	0.000	70.000	0.000	0.000	0.000

Reference Frame Omron TM12 Base **with respect to robot base**

`[X,Y,Z]mm | Rot[X,Y ,Z ]deg - Fanuc/Motoman`

0.000	0.000	0.000	0.000	0.000	0.000

Tool Frame with respect to Reference Frame

`[X,Y,Z]mm | Rot[X,Y ,Z ]deg - Fanuc/Motoman`

-2.443	503.096	673.120	-179.946	-1.466	177.885

Tool Frame

	X	Y	Z
Translation	○	○	●
Rotation	○	○	○

WorkSpace
- ● Do not show
- ○ Show for wrist center
- ○ Show for robot flange
- ○ Show for current tool

Show Frames
- ☐ All/None
- ☐ Base (0)
- ☐ Tool Frame
- ☑ Ref. Frame
- ☑ Robot Flange
- ☐ 1 ☐ 2 ☐ 3
- ☐ 4 ☐ 5 ☐ 6

Joint axis jog

Joint	Value	Min	Max
θ_1	-108.36	-270.0	270.0
θ_2	16.41	-180.0	180.0
θ_3	-97.99	-166.0	166.0
θ_4	-8.88	-180.0	180.0
θ_5	-88.61	-180.0	180.0
θ_6	-16.25	-270.0	270.0

Other configurations ($\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6$)

`( *)-[-108.36°, 16.41°, -97.99°, -8.88°, -88.61°, -16.25°]`

Figure 3.20: RoboDK programming simulator software

Chapter 4

Multimodal HRI strategies

The need for more customized products implies the necessity for machines that are able to adapt and execute the required production tasks in a short configuration time and apply the minimum possible engineering effort to design and program the task. Collaborative robots enable higher safety level and improved flexibility of robotic cells. The latter may be further increased due to the possibility of relying on the operator's input that shares the work area with the robot. Due to these factors, new industrial scenarios and new opportunities to develop more advanced and flexible robotic solutions are continuously being introduced. The use of collaborative robots opens up relevant advantages in the organization of manufacturing industries, allowing the introduction of robots in craft enterprises without the need for major changes in the layout of the production floor. However, the increased deployment of cobots in small and craft businesses is triggering the demand for increasingly simpler and reliable means of robot programming and communication. This is also the case for accurate robotic contact-based operations that require the generation of a precise path to be followed by the robot end effector during the task execution or collaborative assembly tasks. Such duties are required being more agile and re-adapt quickly the robot trajectory to the quickly changing product characteristics.

4.1 Increased safety and flexibility in the footwear industry

The intelligence and connectivity of cobots allow for quick task modifications and the ability to adjust to the flexibility that is required by today's markets, without necessitating extensive modifications to the existing production floor layout. The shoe production industry, which is heavily reliant on manual labor, as highlighted in [75], particularly in smaller craft businesses, is a suitable example to demonstrate the advantages of using cobots and to highlight shortcomings.

Despite the introduction of machinery to aid production processes, human oversight and participation continue to be indispensable in numerous scenarios. This imperative role for human labor comes from the artisanal nature of the footwear industry, which requires rapid adaptation to the challenges presented by globalization and the evolving preferences of discerning consumers. The introduction of traditional automation would imply a reduction in the number of human operators, with consequent strong implications for production times and safety, cost reductions, and productivity improvement, but would scarify most of the flexibility.

The footwear sector is characterized by both functional and fashion objectives, in fact, a shoe must perform well in terms of protection, comfort, and style. Although leather has been the main raw material in earlier styles, many other types of material can be used to enhance any specific shoe objective. To cover a wide variety of types and styles, while offering a product desired by the market, a growing number of big brands and start-ups are relying on the potential of mass customization strategies for footwear [76]. This means that the flexibility of production must be guaranteed, usually with the side effect of increasing the complexity and cost of the production plants.

In the last two decades, many authors have introduced solutions for the automation of parts of the shoe manufacturing process. Some examples can be found in [77] for lasting operation, in [78] for grinding, in [77, 79, 80] for roughing, in [30, 31, 81] for sole bonding, in [82] for sole grasping and in [83] for the finishing operations. In [84], Dulio et al. suggest an automated production system for mass custom production. Furthermore, [75] highlights the benefits of integrating design and production processes.

The coupling between the shoe upper and the sole is one of the last steps in the production process and is often considered the most difficult part of the process [85]. The two sections of the shoe are joined using a variety of methods, depending on the type of shoe that is being made and the tools and technology available for the task. The shoe industry utilizes three primary assembly methods, as outlined in [86].

- cementing: the upper body and the lower part are assembled together using adhesives;
- injection: the sole is injected into a mold, in direct contact with the upper part;
- stitching: the two parts are assembled using threads.

The cementing process has several benefits, such as providing a more flexible and even joint, improved aesthetics, and potential for automation. However, as discussed by the authors of [87], cementing activity is recognized as one of the most critical operations in shoe manufacturing. The assembly process requires the prior preparation of the materials, the cleaning and treatment of the surfaces to be joined, and the preparation of the adhesive solutions. The upper part of the shoe is usually pulled over the last (a model of the shape of the foot) and attached to the sole through an assembly process [88]. Most state-of-the-art papers dealing with the issue of automation of glue deposition, such as [30, 31, 32], exploit a non-contact deposition system based on a robot-moving spraying gun, with the workpiece placed on a worktop. Some alternative solutions, applied in other fields than the footwear industry, as described by [89], rely on gluing systems with a fixed syringe, while robot-moved workpieces have also been proposed.

This section examines the potential of an automated cell for applying glue to a shoe upper, which utilizes a novel method of glue deposition compared to existing techniques. This robotic cell is designed to utilize an extrusion system that is comparable to those used in Fused Deposition Modeling (FDM). The system will be used to deposit molten material, which is initially in the form of a filament. Cell layout, hardware, programming software, and the possibility of developing collaborative applications are presented. Two cell solutions are designed and tested. In the first, the extruder is the robot end effector, whereas the upper of the shoe is grounded to the cell frame. In the second, which is reciprocal, the shoe last is clamped to the robot wrist and the extruder is fixed to the cell frame. The glue deposition trajectories are determined using the upper shoe CAD model, allowing control over the inclination of the extruder nozzle in relation to the surface normals. The use of CAD models as input data for trajectory planning allows for quick changes in the robotic application by updating the input. This not only speeds up the process update but completely cancels the necessity for robot re-programming in case of product variation. Both cell layouts presented yield commendable quality and satisfactory production times. The peculiarities of the two solutions are highlighted and compared in terms of cell layout and the possibility for the robotic cell to be adapted to a wide variety of processes. In particular, the cell configuration with the fixed extruder minimizes grip instances and prioritizes safety since the stationary extruder can be placed in a restricted region outside the collaborative workspace. This design promotes secure operator interaction within the collaborative working area of the machine, while also facilitating the creation of a user-friendly robotic cell. Consequently, it lays the foundation for the implementation of automation in craft-oriented environments.

4.1.1 Robotic cell layouts

In order to validate different approaches and compare them to highlight the benefits of each, two cell solutions are discussed:

- Mobile extruder cell type: the last is anchored to the ground and the robot holds a mobile extruder for glue deposition;

- Fixed extruder cell type: the extruder for glue deposition is grounded to a fixed frame, and the shoe last is anchored to the robot and moved to reach the relative desired poses.

The initial element to consider is the practicality of implementing the corresponding robotic cell design. This involves the challenge of controlling the hot extruder with a robot in the first place or using the robot to hold the shoe last in the second. The two designs necessitate two distinct approaches to utilizing robotic resources: when a mobile extruder is employed, the robot is devoted exclusively to a single task and must be reconfigured to carry out different processes. Conversely, when the robot is grasping the shoe last, a variety of processes can be performed with fixed tools (e.g. cementing, roughing, power pressing) without having to remove the shoe last from the robot's wrist.

The robotic cells proposed for the automated deposition process are made up of two main subsystems:

- A system has been developed to deposit molten material in a synchronized manner with the robot's movement, allowing for the precise amount of adhesive to be applied.
- A 6-degree-of-freedom manipulator is used to accurately position the system for material deposition in relation to the shoe upper.

Figure 4.1 represents the two proposed alternative configurations, which aimed to regulate the relative position and orientation between the glue supply system and the 3D-shaped surface of the shoe. Both use the collaborative Techman Robot TM5-700 manipulator. The setup represented in Figure 4.1a is based on a mobile extruder and an orientable nozzle that is directly connected to the robot wrist. The shoe is kept firmly on the ground in the operational area. In comparison to the first configuration, the second one is based on a stationary nozzle that is displaced in the working area, with the robot wrist grasping and manipulating the last and the shoe upper placed on it. This setup is illustrated in Figure 4.1b. The extruder is held by a supporting structure on top of the working area, whereas the last is clamped by a 3D printed fastener connected to the robot wrist.


(a) (b)

Figure 4.1: Experimental test layouts: a) Mobile extruder; b) Mobile shoe last

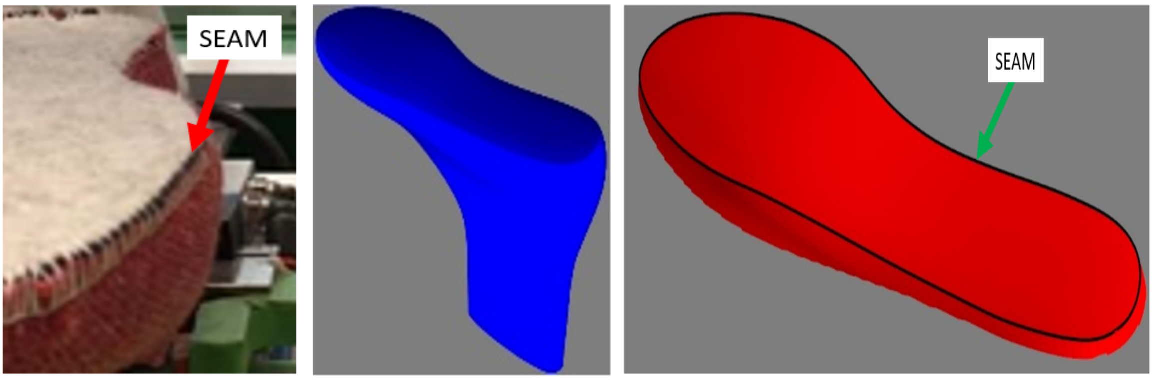
During the glue deposition, the tool supplies a suitable flow of glue at a chosen rate, to deposit the required adhesive film over the curved surface. The glue deposition system described in the previous section 3.2.2 provides the flow of adhesive material to be deposited on the upper of the shoe. The trajectory of the manipulator is designed based on a CAD model of the shoe upper and a slicing software created for this purpose. The latter generates the path to follow on the upper surfaces. The software outputs a set of poses (positions and angular orientations in the space) to be followed during glue deposition on the shoe upper. Once the path has been defined by the off-line procedure, the robot motion has to be designed to ensure that contact is maintained during the trajectory execution. For the mobile extruder case, the nozzle is considered as tool center point (TCP) of the robot, which can consequently be easily programmed to make the end-effector get in contact with the upper surface with the desired orientation. On the other hand, an alternative algorithm is adopted for the movable last: the surfaces of the shoe upper, on which the fixed nozzle has to draw the path, are moved in three dimensions together with the last. All the output poses of the slicing software are first referred to a single point of the last to create a TCP reference, whose distance with respect to the clamp and consequently to the robot wrist is known.

During the glue deposition process, a proper contact force must be maintained between the extruder nozzle and the upper surface. Inaccuracies in the shoe upper CAD model or misalignment in the last clamp, which can lead to detachments or sudden increase in the contact force, resulting in a nozzle closure, are tackled by using a force feedback control strategy. For these reasons, the proposed robotic cell exploits a force feedback control, in parallel to the control position: the tentative trajectories initially defined off-line are adjusted online through a force control with constant reference force along the direction of the nozzle axis. The load cell described in 3.3.1, is mounted between the robot wrist and the extruder, in the movable extruder setup (Fig. 4.1a), and between the robot wrist and the shoe last, in the fixed extruder setup (Fig. 4.1b).

4.1.2 Path generation using CAD model

In the literature, several methods have been reported that can be exploited to extract the path that the robot should follow from the CAD model, using slicing software, as described in [90, 91]. The slicing software used in his work provides a six-dimensional path characterized by the position coordinates (x, y, z) and the components $(cos(\xi), cos(\psi), cos(\phi))$ of the unit vector normal to the surface at each point, defined in an absolute reference frame. The robot poses, and thus the joint coordinates to generate the desired trajectories on the shoe upper, have to be derived. This step is straightforward in the case of the mobile extruder solution, where the TCP, defined as the tip of the nozzle, must assume the same poses with the same orientation, as presented in [92].

On the other hand, in the case of mobile shoe with the last clamped to the robot wrist, the path output by the slicing software must be referenced to a unique frame belonging to the shoe last, whose origin's position and angular position with respect to the robot wrist are known, while additional geometrical transformation must be applied. It is possible to extract the mesh representation, in terms of connectivity list and vertices, starting from a CAD model in STL format. The upper of the shoe, the STL model of the shoe last and the selected portion where the glue must be placed are represented in figure 4.2.



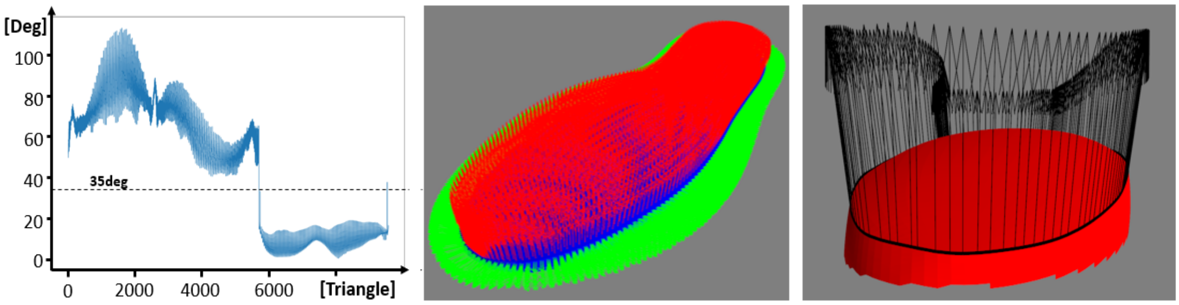
(a) (b) (c)

Figure 4.2: a) Shoe upper b) STL file of the shoe upper c) Reference glue deposition trace.

As can be observed in the upper shape reported in Figures 4.2a and 4.2c, the seam corresponds to an abrupt variation in the normal vector to the surface. Therefore, normal vectors can be obtained from the definition of each triangle of the mesh, where a sudden variation occurs in the normal vector direction n_i. For each triangle, the angular distance between the normal vector n_i and the unit vector that defines the vertical reference direction n_{ideal} can then be computed using the inner product

$$\theta_i = \arccos(n_i \cdot n_{ideal})$$

Figure 4.3a represents the trend of the angle θ obtained for the shoe upper considered. Two groups of angles can be identified. One in the range of 50^o to 100^o degrees, corresponding to the mesh of the side surface of the upper, and a second in the range of 0 to 20 degrees, corresponding to the mesh triangles belonging to the upper surface of the shoe. A threshold value can be defined (e.g.35^o) to separate the mesh triangles belonging to the top surface (red normal vectors in Fig.4.3b) from those belonging to the side surface (green normal vectors in Fig.4.3b). The points defining the contour of the seam can be identified by the boundary points that belong to the top surface. The final seam line is represented in Fig.4.3c, together with the corresponding unit vectors normal to the surface.



(a) *(b)* *(c)*

Figure 4.3: Seam line evaluation. a) Normal vectors obtained from the definition of each triangle of the mesh, where a sudden variation occurs in the normal vector direction. Trend of the angle θ obtained for the shoe upper and a threshold value of 35^o that separate the mesh triangles belonging to the top surface from the side surface are represented; b) In red are represented the vectors normal to the top surface, while in green those belonging to the side surface; c) The final seam line with the corresponding unit vectors normal to the surface.

Once the seam line has been identified, the desired path for glue deposition is calculated by scaling the seam contour inward to generate a concentric perimeter on the upper surface of the shoe (using algorithms derived from computer-aided machining, such as [93]).

4.1.3 Robot motion

The generated path consists of a set of points (position coordinates X, Y, Z), and direction cosines $(cos(\xi), cos(\psi), cos(\phi))$ of the vector normal to the upper surface of the shoe at each point. In the cell layout with the mobile extruder, the EE consists of the extruder itself. The EE frame is defined with the Z axis parallel to the extruder nozzle and oriented toward the shoe surface. Once the upper of the shoe is positioned and grounded to the cell frame, the previously computed path can be referred to the absolute reference frame of the robot base and can be adopted as the starting point to define the poses of the end effector. Based on the geometry of the extruder, the position coordinates are set so that the nozzle comes into contact with the surface of

the workpiece along the deposition path. Normal vectors of the path are used as a reference orientation for the extruder axis.
In the layout with a mobile shoe and a fixed extruder, the EE consists of the gripping system for the upper of the shoe, and its reference frame is defined with a Z-axis perpendicular to the robot wrist flange, oriented outward from the gripper of the shoe and exiting the surface of the shoe. In this case, some additional mathematical steps are required to obtain the robot motion starting from the previously computed path. The procedure to define the EE poses requires an intermediate step in which the path generated by the slicing software is referred to a single reference frame belonging to the shoe last. The latter is firmly connected to the robot wrist, so the position and orientation relative to the robot wrist's reference frame are known. This step allows us to define the coordinates of each point of the deposition path with respect to the reference frame of the robot wrist. The normal vectors to the upper surface of the shoe can then be oriented with respect to the axis of the fixed extruder. The robot must obviously be able to reach the EE pose, and move along the path without colliding with the shoe For a given set of EE poses determined by the slicing process, there are a large number of possible robot movements that can be used to execute the desired trajectory, as a result of the numerous configurations available at each point and the combinations of these configurations. From the numerous solutions available, the feasible subset must ensure that the EE can keep the extruder nozzle in contact with the upper surface of the shoe, while meeting certain criteria, such as:

- **Collision**: the EE must be able to move along the path without hitting the shoe. No contact between objects in the cell is allowed, especially in this application, where hot surfaces could harm the robot and electrical cables.

- **Range of motion**: the EE pose must be accessible by the robot. In addition to the need for a point to be reachable, the rotation around the tool axis might be restricted for a given machine. Therefore, the paths that lead to continuous rotation of the wrist joint $J6$ through consecutive points must be avoided.

- **Tool cabling**: Tool wires can limit the range of motion of the robot due to their pulling action or the need to prevent excessive twisting.

Among the various approaches that can be adopted to define an EE orientation for each point in the path, in this work, the "Rodrigues rotation formula" is exploited. Its solution computes the rotation matrix to align each upper surface normal vector with the Z axis of the tool. Schematic cell layout is shown in Figure 4.4: the absolute reference frame is reported in red on the robot base frame; the vector p_k represents the position vector of the tip of the extruder nozzle, p_{ee} the position of the EE reference frame, and p_{tool} the position of each point of the path on the shoe relative to the reference frame of the tool.

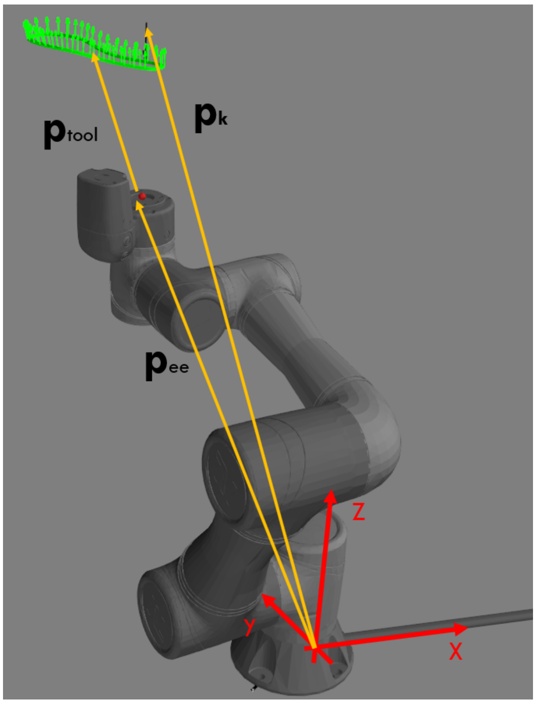


Figure 4.4: Scheme of the mobile upper layout for defining the EE poses

The figure in reference 4.5a shows the starting position of the system, with the vector v perpendicular to the surface of the shoe that is not yet aligned with the extruder axis V_{target}. The robot wrist reference frame is colored blue. The vector p describes the relative positions of the deposition path points with respect to the relative frame of the robot wrist. When considering an initial vector $\boldsymbol{v}$ in $\Re^3$ and a unit vector $\boldsymbol{w}$ defining the rotation axis around which $\boldsymbol{v}$ rotates by an angle θ_r, the rotation matrix can be calculated as follows.

$$\mathbf{R} = \mathbf{I} + (\sin\theta_r)\mathbf{K} + (1 - \cos\theta_r)\mathbf{K}^2 \tag{4.1}$$

so that the new rotated vector is

$$\mathbf{v_{rot}} = \mathbf{R}\mathbf{v} \tag{4.2}$$

Both the rotation axis $\boldsymbol{w}$ and the rotation angle θ_r are the unknown variables to be evaluated in the presented application. v is the initial vector normal to the shoe upper surface and v_{rot} the final vector oriented as the extruder axis (or at will with respect to it). The term $\sin(\theta_r)\mathbf{K}$ of equation (4.1) can be determined as:

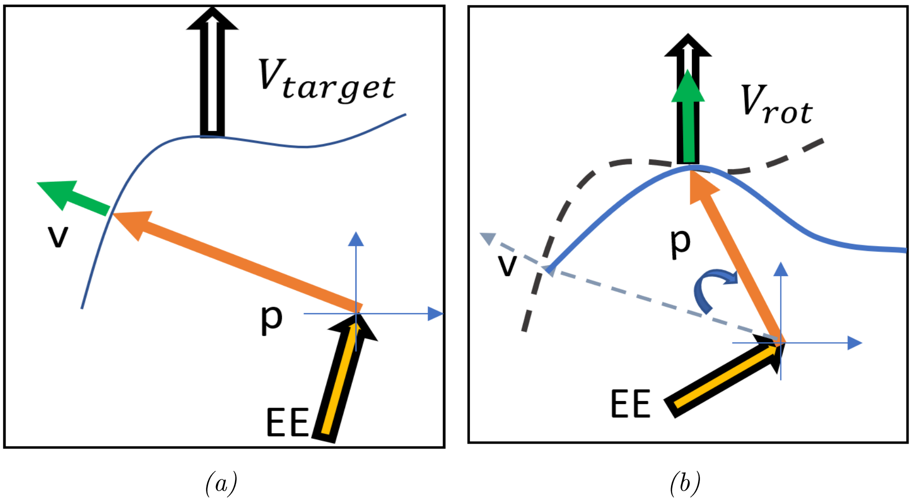
$$\sin(\theta_r)\mathbf{K} = \begin{bmatrix} 0 & -k_z & k_y \\ k_z & 0 & -k_x \\ -k_y & k_x & 0 \end{bmatrix} \tag{4.3}$$

where the elements of the matrix are the components of the k vector, computed as:

$$k = v \times v_{rot} \tag{4.4}$$

and oriented as the unit vector oriented $\boldsymbol{w}$.
The remaining term to be computed in equation (4.1) is $\cos(\theta_r)$, which can be

obtained through the dot product between the normalized vectors v and v_{rot}. The rotation matrix bringing v into v_{rot} is then computed substituting the calculated terms in equation (4.1). A dedicated rotation matrix must be calculated for each of the points that define the deposition path along the upper surface of the shoe, but all these steps can be defined in the cell layout design stage, provided the CAD model of the upper shoe is available.



(a) (b)

Figure 4.5: Rodrigues' rotation algorithm representation

After the EE poses generating the desired tool path are fully defined the joint coordinates can be computed to check their feasibility.

This procedure is highly adaptable for any changes that need to be made to the upper, last, holding support, or nozzle to accommodate a new shoe model. Only the geometric input data of the algorithm need to be altered without requiring advanced programming knowledge. The solution is also general enough to be applicable to different shoe models, which is a great achievement in terms of the proposed automated cell's flexibility. Particular attention should be paid when selecting a solution that does not cause hazardous twisting of the cables connected to the extruder, the load cell, and the extruded glue filament. In this regard, the cell layout with a fixed extruder is the best choice. After discarding the solutions that are not feasible from a kinematic point of view, the ones remaining need to undergo a collision and self-collision check, before being validated.

4.1.4 Experimental results

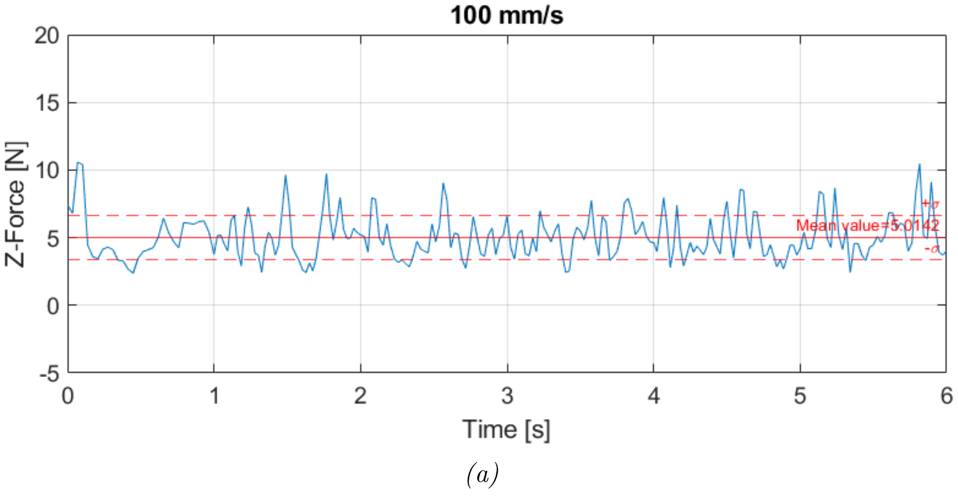
The objective of the experimental phase is to assess the ability of the developed system to guarantee a regular deposition of glue on the upper surface of the shoe, along the designed trajectory. Based on the specifications and preliminary tests of the glue manufacturer, a target contact force is established at $5N$. In any case, good quality of glue deposition is expected, provided that the contact force is within the range $2-10N$ [94], so that the experimental tests are aimed at verifying the absence of severe contact losses or peaks in force.

A deposition speed of around $100mm/s$ could guarantee satisfactory cycle times, which is a good compromise between the need to obtain the lowest possible cycle time and the ability of the built-in robot control to actually maintain proper contact force on the 3D surface of the shoe. The latter becomes more challenging as the Tool Contact Point (TCP) speed increases. For these reasons, experimental tests investigated the resulting contact force with a linear TCP speed ranging between $50mm/s$ and $200mm/s$. The highest limit shall achieve the maximum speed prescribed by ISO 10218 [95] standard for any robot operating in the proximity of a person (i.e. $250mm/s$), while the lower limit is selected to obtain a low speed case to be used as a reference for comparison with higher speeds. In the entire speed range considered, the regularity of the glue flow shall be guaranteed.

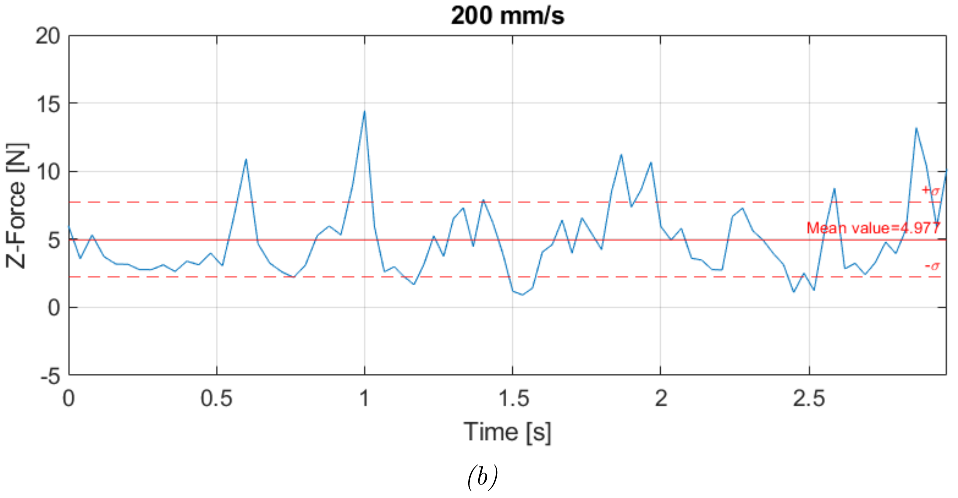
For the glue deposition process in the footwear industry, a positional error of up to a few millimeters can be accepted, according to [30] and [96], so that the performance of the robotic cell can only be evaluated by measuring the contact force between the nozzle and the upper surface and by eye inspection. The collaborative robot TM5-700 exploited in the tests has repeatability equal to $\pm 0.05mm$. However, at this stage of the research activity, no visual system has been used to verify the geometry of the dispensed glue strip.

Mobile extruder

Figure 4.6 reports the contact force results achieved with the mobile extruder moved by the collaborative TM robot at a speeds of $100mm/s$ (Figure 4.6a) and $200mm/s$ (Figure 4.6b).



(a)



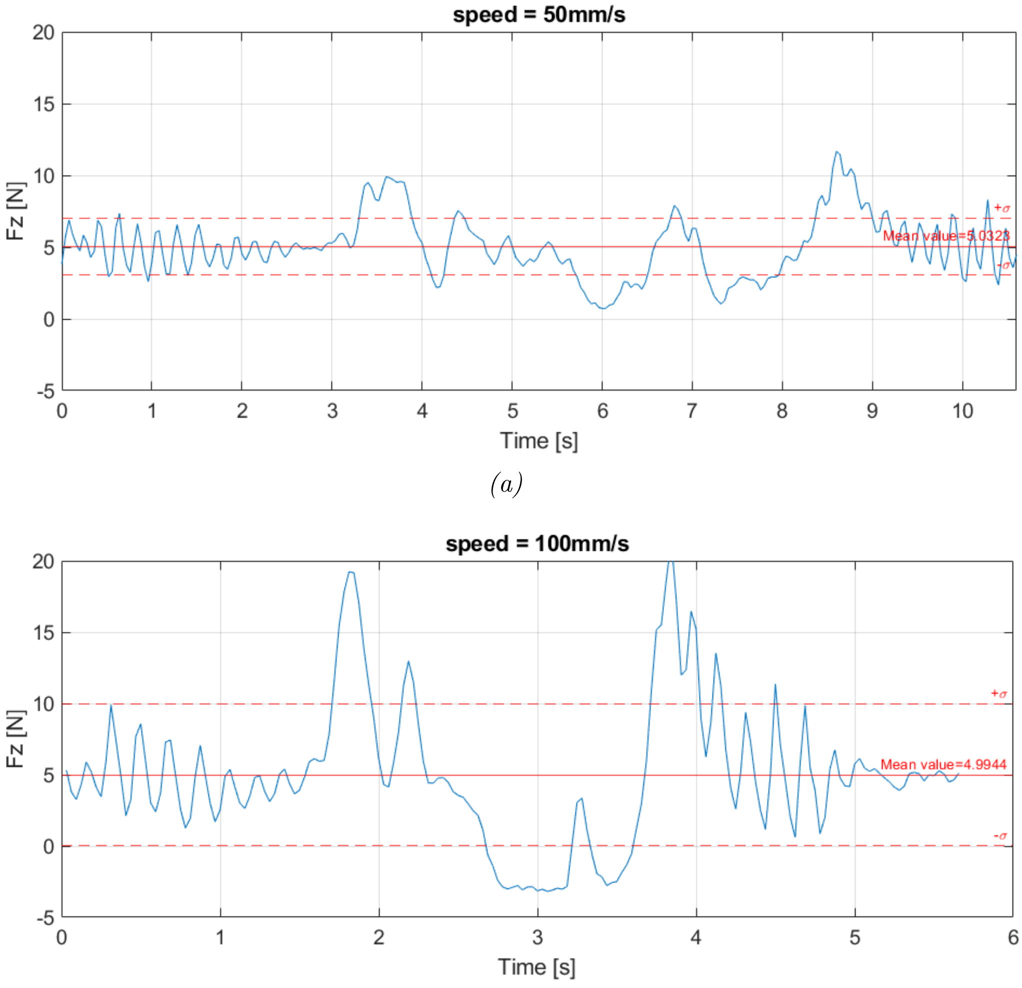
(b)

Figure 4.6: Contact force results for the cell layout with mobile extruder. (a) $100mm/s$. *(b)* $200mm/s$.

The experimental results showed a full capability of the system to obtain the linear speed of 200 mm/s, whereas, when dealing with the capability to maintain the proper contact force, the results show an increase in the standard deviation of the contact force starting from the speed of 100 mm/s. Contact loss is defined as the percentage of samples in which the contact force decreases to less than 10% of the average target force (i.e. $F < 0.5N$), on the total number of samples. Despite the fact that the standard deviation of the contact force increases with increasing speed, a low percentage of contact loss (0.2%) is detected at a speed of 200 mm/s. As recalled in [94], the presence of short detachment does not affect the possibility of depositing the melted glue on the upper surface of the shoe.

Fixed extruder

In the case of a test cell with a mobile shoe upper and a fixed extruder, in which the shoe last is directly attached to the robot wrist, the control performance is expected to be intrinsically worse due to the higher inertia attached to the robot wrist and to the fact that the moments applied to the robot wrist rapidly vary as a consequence of the relative position between the extruder tip and the upper surface of the shoe. With reference to the setup represented in Figure 4.1b, in fact, for a given contact force between the extruder and the upper surface of the shoe, lower torques are generated when the extruder is in contact with the heel rather than when the contact is with the tip. Figures 4.7a and 4.7b report as an example the results achieved with the TM robot at a speed of 50mm/s and 100mm/s, respectively. In Figure 4.7b detachments are observed around 3 s, corresponding to the sharp curve zone of the upper toe of the shoe.



(a)

(b)

Figure 4.7: Contact force results for the cell layout with mobile shoe upper and fixed extruder. a) 50 mm/s (4.7a) b) 100 mm/s (4.7b).

The results of the experimental phase confirmed that it is feasible to reach a maximum deposition rate of 200 mm/s in the mobile extruder cell layout, with a positive force or a restricted contact loss rate. Cell layout with fixed extruder has the potential to decrease the number of grips, thus shortening the total cycle time and improving the safety of collaborative tasks. When the upper of the shoe is held directly by the robot, indeed, right after the glue deposition phase, it could be driven directly to the press for sole fastening, without the need to regrip the workpiece. Furthermore, when the robot moves the upper of the shoe and the hot extruder is fixed, the robotic cell could be upgraded more easily to become a collaborative robotic cell, as explained in [97, 98], due to the fact that this solution does not have hot mobile parts in the operational space, allowing a higher degree of security for operators. The use of collaborative robots opens up relevant advantages in terms of improving the organization of manufacturing industries, allowing the introduction of robots in craft enterprises without the need for major changes in the layout of the production floor.

4.2 Trajectory planning by use of a 3D stereo depth camera

The trajectory of the glue deposition path was calculated using data from the CAD model, which required force feedback to ensure the correct deposition. To counterbalance any discrepancies between the digital shoe last model and the shoe upper surface, force sensor measurements were used as feedback to compensate for any potential positioning errors along the deposition axis. The force between the extrusion tool and the shoe upper was measured and used in force control mode to investigate the possibility of maintaining an appropriate contact force between the extruder and the shoe upper. Errors are likely to occur during the trajectory planning process, as the physical upper of the shoe, composed of various materials with distinct characteristics and manually placed on the shoe last, does not accurately correspond to the shape of the CAD model. Moreover, in craft businesses, it is common for the CAD model to be not available and the shoe to be custom-made, making it impossible to carry out the proposed trajectory planning process.

This section describes a procedure for automating the generation of trajectories based solely on the use of a vision-based approach combined with force feedback, thus avoiding the extraction of data from the CAD model. Force-feedback control is used in parallel to position control to compensate for inaccuracies in the generated trajectory, which may be caused by the resolution of the vision system. This could guarantee the proper execution of industrial tasks like deburring, roughing, and gluing by means of a robotic solution for contact-based industrial operations on irregular curvilinear shapes. To this end, the end-effector trajectory can be created using data obtained from a 3D stereo depth camera, without any prior knowledge of the object's shape and taking advantage of both the depth and color data collected by the vision system. In the following subsection 4.2.1 the proposed solution is explained, focusing on the computer vision techniques used and the details of the algorithm developed. In subsection 4.2.2, the cell setup used in the experiment is outlined and the results are discussed to reach the conclusions of subsection 4.2.3.

4.2.1 Path generation using vision system

In this section, the proposed solution is to use a RealSense D435 3D stereo depth camera, which is outlined in 3.3.2, to define the path for glue deposition. This device is favored for its performance and affordability. The SDK (Software Development Kit) that comes with it allows for the development of solutions in the most popular programming languages (C/C++, Python and MATLAB) and easy integration with ROS (Robot Operating System). The camera is capable of producing depth information with a resolution of up to 1280p × 720p through the use of stereo vision depth technology. This method makes use of two image sensors to capture the scene in the camera's field of view from two distinct angles. By comparing the positions of the same points in the two images and taking into account the data related to the position and focal length of the two sensors, the depth can be determined. Additionally, the camera is equipped with a separate RGB sensor with a resolution of up to 1920p × 1080p. An alignment procedure is required to match the pixels with the same coordinates in the color and depth frames, since they are generated by two distinct sensors with different perspectives and possibly different resolutions.

Texture mapping and alignment are essential because of the various sensors used to capture color and depth images. This leads to different positions, orientations, and intrinsic calibration parameters (e.g. width and height of video stream, principle point, focal length, lens distortion coefficients). As a result, it is not possible to directly link the same pixel coordinates in the depth and RGB images. Texture mapping is used to assign the RGB value recorded from the RGB sensor to each point in the depth-point cloud. This is done in the following way:

1. Deproject points of the depth frame from camera pixel coordinates (with the origin at the top left of the image) to 3D world coordinates (defined in meters with respect to the physical sensor center). This step is done using the intrinsic calibration parameters of the sensor. The result is a point cloud.

2. Applying to the resulting point cloud the extrinsic matrix (made up of the rotation matrix R and the translation vector t between the depth sensor and the RGB sensor), it is possible to calculate the 3D coordinates of the points with respect to the reference frame of the RGB sensor (the RGB sensor is becoming the new origin).

3. UV-map: project points from world coordinates to color camera pixel coordinates (RGB sensor). This is done by using intrinsic parameters.

4. Alignment: To correlate pixels in the depth and color images it is necessary to create an artificial view that shows how a color stream would be if it were captured from the depth sensor perspective or vice versa. In this work, the depth frame is aligned with the color frame.

Once the coordinates of color and depth frames have been established in relation to the same reference frame, it is possible to process their data simultaneously. This algorithm, written in Python and utilizing image processing from the Open-source Open-CV and Librealsense libraries, is composed of the following steps:

1. Remove background: In the industrial setup of the gluing process, the shoe is hold in the desired position, the distance from the 3D camera and the shoe surface is known within a tolerance range. By setting a distance threshold, pixels in the background can be removed from both the depth and the color images.

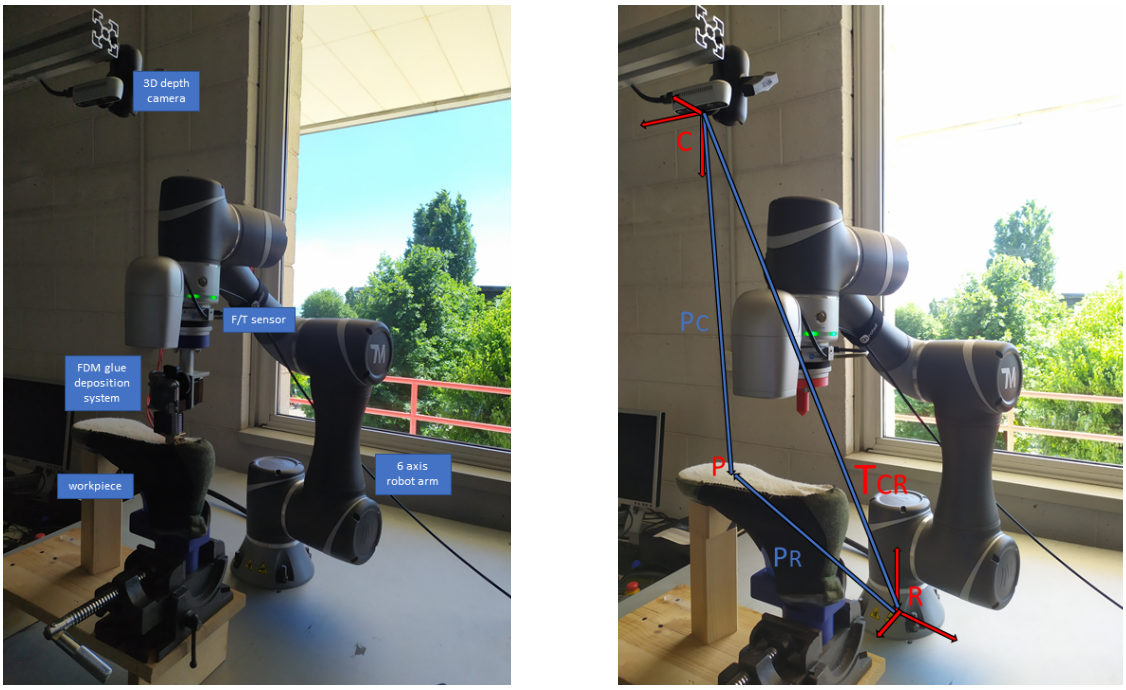
2. Search for the object of interest and glue deposition area: the shoe model considered in this work is made by a shoe upper of white color and a stitching area surrounding it. The developed searching algorithm consists of searching for the area in which a great color variation occurs, in the case considered between the white color of the shoe upper and black stitches and side parts. The same algorithm can be used for different shoe sizes and models, by only changing the color to detect the upper area of the shoe. The glue has to be deposited following the stitching pattern with an offset toward the center of the shoe upper.

3. Obtain the pixel coordinates of a possible gluing path: to generate a suitable gluing path, a contour finding algorithm identifies the contour surrounding the interested area that coincides with the stitching area. The OpenCv library is used to carry out the contour-finding process. This library enables the development of image-processing algorithms in the most popular programming languages, such as C/C++, Java, and Python. The algorithm consists in the transformation of grayscale image from the filtered color image obtained from the first step into binary image using a color threshold. In this way, the pixels of the interest area, having white color, will have a one value and zero values will be assigned to the other pixels. Using the OpenCV.findContour() function, the contour is calculated. Due to imperfections in the grayscale image, more than one contour may be detected. The OpenCV.contourArea() function is used to detect the largest contour related to the object considered.

4. The contour found is scaled toward the center of the upper surface of the shoe to generate the path.

After determining the pixels that form the gluing path to be followed by the glue deposition system, it is necessary to transform the points that are now in the camera coordinates from pixels to millimeters. This task is accomplished by utilizing the camera parameters given by the producer and is obtained through the specialized function of the Librealsense SDK. The path then has to be defined with respect to the robot coordinate system. A calibration procedure is performed to calculate the homogeneous transformation matrix T_{RC} to apply the required transformation as shown in the setup in Figure 4.8. The position of at least one point is determined in both the reference frames of the coordinate system (camera and robot base) to create the homogeneous transformation matrix (rotation matrix and translation vector between the two reference frames). The point used in the calibration process is a point with different colors placed on the shoe surface whose pose with respect to the camera reference frame is detected by a color-search-based algorithm. To define the coordinates of the same black point with respect to the robot base reference frame, the robot is moved manually until a pointed tester used as the robot's end effector reaches the point position.

The experimental setup of the robotic cell shown in Figure 4.8 was developed to test the quality of the results obtained. A force/torque sensor is placed between the robot and the glue extruder to allow a force control loop like the one described in the previous section 4.1. The setup consists of: a six-axis Techman TM5-700 collaborative robot, a RealSense D435 stereo depth camera, an Onrobot HEX-E force / torque sensor, and a 3D printing FDM system for glue deposition. The considered depth camera has an optimal working range of distance from the workpiece that is between $0.3m$ and $3m$ and a minimum distance of $0.28m$. The depth error increases with the working distance, being less than 2% at the distance of $2m$. The shoe is held with the upper part at a distance of $0.3m$ from the 3D camera, and the most distant point captured is at a distance of $1.70m$. Different experiments are carried out to test performance under different conditions, changing the robot speed with values of 25mm/s, 50mm/s, 75mm/s and 100mm/s, and with a contact force set point equal to 5N, as reported in section 4.1.



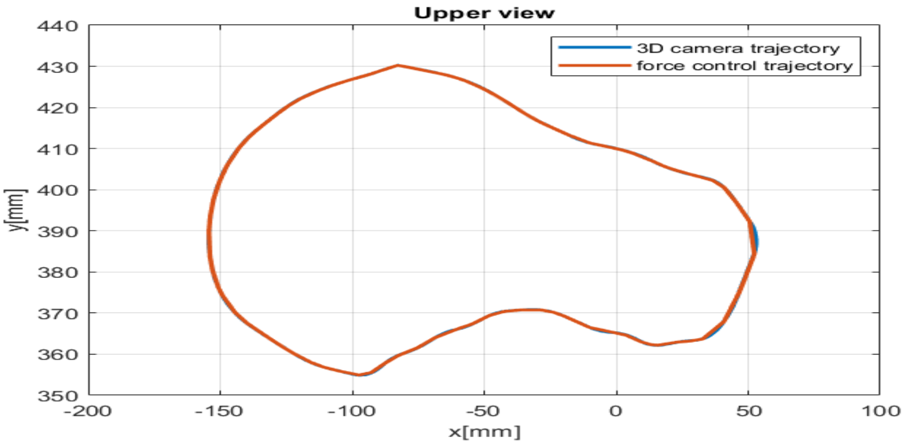
(a) *(b)*

Figure 4.8: Experimental setup a) Cell layout b) Reference frames transformation

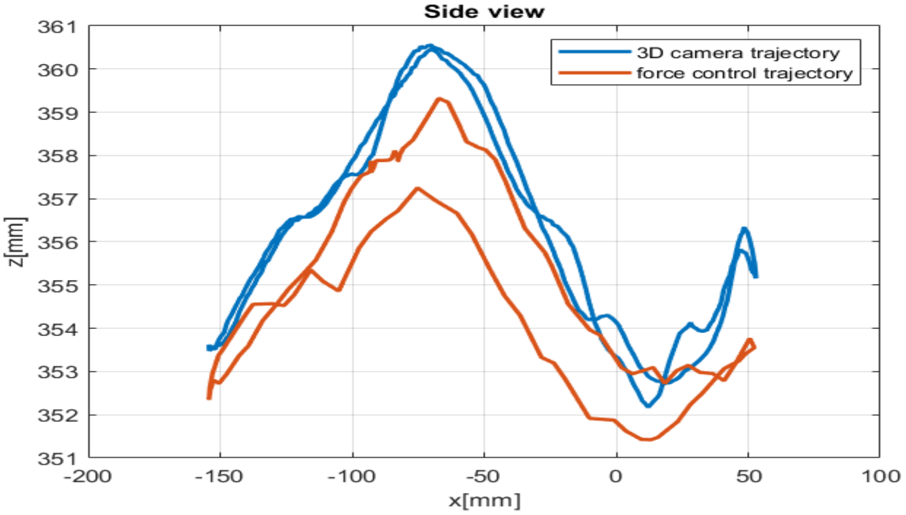
4.2.2 Experimental results of the vision-based trajectory planning

The measured contact force, adopted as the evaluation parameter for the assessment of the process, is strictly related to the quality of the trajectory sent to the robot. Trajectory errors or imperfections, such as a lack of smoothness, can indeed cause a worsening of the performance of the force control loop, causing contact loss or excessive oscillations in the contact force value. In order to enhance the trajectory input, it is essential to eliminate points that are too close to each other (e.g. less than 10 or $5mm$) after the conversion from pixels to millimeters. This will enable the robot controller to interpolate the trajectory more accurately, thus improving the performance of the force control loop.

To evaluate the trajectories defined through the visual system, as depicted in Figure 4.9, we compare the trajectory generated through the 3D camera to the one effectively followed by the robot TCP, exploiting the adjustments of the force control loop at a speed of 100mm/s. Figure 4.9a refers to the top view in the XY plane, while Figure 4.9b refers to the side view.



(a) Upper view



(b) Side view

Figure 4.9: Trajectory comparison

As can be seen in Figure 4.9b, even when the points are filtered to improve the reference trajectory, there is still a noticeable difference between the two trajectories in the z direction. The force control loop is advantageous in this application as it compensates for any errors in the 3D vision system caused by the material being compressed under the contact force. Figure 4.10 reports the force measurements achieved for different linear speeds of the robot TCP, where the mean value is close enough to the target value of 5N. At low speed ($25mm/s$) the mean value of the contact force is equal to $4.98N$ and the standard deviation is $1.02N$. When increasing the speed to $100mm/s$ the mean value is $5.01N$ and the standard deviation of the force equals $2.03N$. These force values are comparable to those reported in the previous section 4.1, in which the TCP trajectories were defined on the basis of the CAD model of the shoe upper.

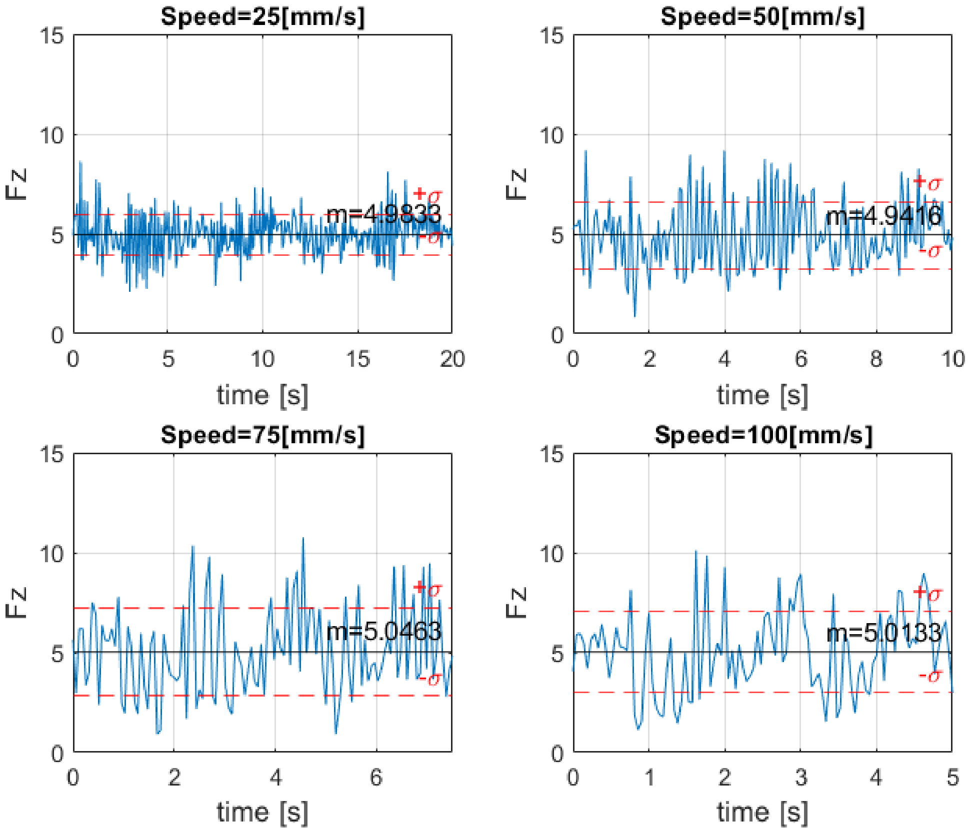


Figure 4.10: Contact force at different speeds

Finally, Figure 4.11 shows the contact force results corresponding to the cases in which the first trajectory obtained by the vision system is filtered with the distance set to 5 and $10mm$ between each point. Results refer to a TCP speed of $50mm/s$. It is possible to observe that the increase of the filtering distance allows a reduction of the force peak value from $12.3N$ to $9.54N$, as well as a lower standard deviation (from $1.84N$ to $1.8N$).

4.2.3 Final considerations

This section presented a solution for automated trajectory planning of contact-based industrial tasks using an Intel RealSense 3D stereo depth camera, a 6-axis collaborative robot, a force / torque sensor and an FDM glue deposition system. This study has indicated that a 3D stereo depth camera and a force/torque sensor

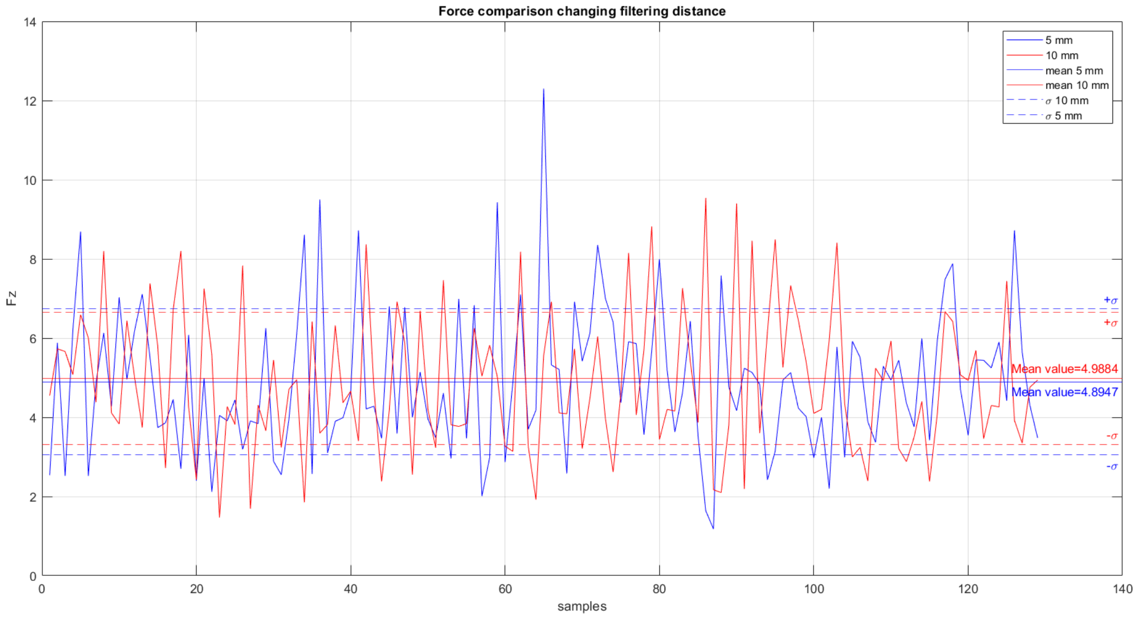


Figure 4.11: Force comparison between different filtering distances

are suitable for the automated trajectory planning of contact-based activities, where a precise trajectory and an appropriate contact force must be ensured. The proposed solution was applied in the application of glue deposition on a non-flat workpiece without prior knowledge of the workpiece features. The system was able to successfully execute the glue deposition process at various speeds while maintaining satisfactory contact force behavior.

The results of this study demonstrate the potential of vision-based trajectory planning to increase the flexibility of contact-based applications, allowing the production process to quickly adapt to any product changes. Among all the processes necessary to assemble a shoe, the deposition of glue on the shoe upper is one of those that requires safety improvements to allow the operator to be anywhere near the workstation. In fact, if the right safety precautions are taken, the alternative arrangement of transporting the shoe last instead of the hot extruder allows a human to be present in the robotic cell. This provides a wide range of potential modifications to the assignment of tasks throughout the assembly process, allowing the operator to monitor or adjust production while it is in progress. However, to fully exploit the collaborative nature of the TM5-700 it is necessary to consider the transitions between various robotic and human tasks, improving the interaction strategies.

4.3 Brain computer interface for human-cobot interaction

With the advent of industry 4.0 technologies and smart systems, industrial workstations and work areas have evolved to meet the demand for high-performance production and organizational flexibility at the same time. Robotic installations easily meet these needs, so their use is increasing rapidly in industry, as reported in [99, 60]. However, robots still struggle if compared to human workers innate ability to adapt to unexpected events and maintain strong decision-making skills, even in a dynamic and complex environment. Where sophisticated tasks still require human intervention, collaborative robots are employed to cooperate with humans to improve the efficiency and productivity of the workstation.

Cobots are user-friendly tools that can potentially increase productivity while still maintaining the advantages of a human-centered system. These machines are compact and can be programmed to carry out a variety of tasks, making the workplace of employees more efficient. However, the utilization of cobots in industrial settings is still largely determined by the approaches used for robotic cells that are not designed for collaboration. Robots that work together with humans are usually used with safety regulations (e.g. limits on forces and speeds) and flexibility needs (the capacity to be quickly and easily reprogrammed) in mind, not taking in account the possibility of the robot to interact and accept inputs from the worker, as discussed in section 4.1. The risk is to reduce collaborative manufacturing solutions to traditional robots with the addition of safety limitations, as warned in [100].

Recent studies are exploring the genuine "collaboration" and mutual aid between the human operator and the cobot as possible improvement of collaborative workstations. Scientists are working to discover ways to enhance the interactions between the cobot and the operator, making communication and programming simpler. To create a practical collaborative application, which is also interactive, it is necessary to have a well-thought-out task design and an effective communication plan between a human and a cobot. The achievement of a successful and productive communication strategy between humans and cobots is a major accomplishment of collaborative approaches that can be based on a variety of communication technologies, potentially including multiple modes. Humans have flexibility, adaptability to learn new tasks, and intelligence, while robots can guarantee physical strength, repeatability, and accuracy.

In this section, a task-oriented algorithm is proposed to enable smooth interaction through the use of biosignals collected by a Brain Compute Interface (BCI), described in section 3.3.3, as a way to facilitate collaborative workstations. The goal is to provide an implementation method of BCI devices for industrial applications that makes cobots easy to interact with. The proposed solution is tested in an experimental setup that uses the Steady-State Visual Evoked Potential (SSVEP) response signal to drive machine commands to a collaborative robotic arm.

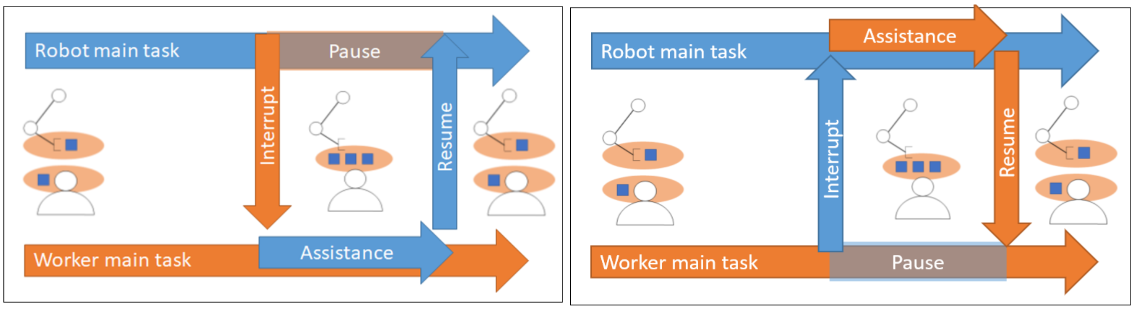
The adoption of a wearable device for the acquisition of EEG signals enables a flexible, hands-free collaboration and provides a valid alternative to the other methods, such as those described in section 1.2.2, overcoming some serious limitations of such technologies in industrial environments. Light conditions have a strong impact on the effectiveness of computer vision systems and, in case the gesture recognition approach is used, the operators must be trained in advance, whereas speech recognition is unusable in noisy environments. A traditional teaching pendant controller, even if it is surely robust, does not provide hands-free intuitive interaction. The potentiality of BCI techniques in industry 4.0 applications has been highlighted in [101], being widely studied in the field of neuroscience [102, 103, 104].

This section presents a human-robot interaction strategy that enables multitasking. This approach allows BCI devices to be used in industrial collaborative workstations, making it easier for cobots to interact. The concept behind the proposed solution is explained and the experimental setup outlines the tools used to evaluate the suggested algorithm.

4.3.1 On-demand assistance

Operating in the same environment and potentially on the same object, the ability to communicate with the cobot without halting the primary program and having to reprogram it would be a great benefit in terms of cycle time and usability. As reported in section 1.2.2, a BCI using steady-state visual evoked potentials (SSVEP) can reach high accuracy for command recognition. A subject wearing a BCI headset can provide input to a robot by looking at a flickering stimulus on a light source, such as a screen, that is running SSVEP stimulation. To classify electroencephalography (EEG) data, Fast Fourier Transformation (FFT) with linear discriminant analysis (LDA) can be used. SSVEP-based BCI provides accurate and efficient high-level control of a robotic arm, allowing a BCI-based control system for operational assistance.

To make the cobot a useful collaborative tool, the worker should be able to ask for help in completing their tasks. In the dual situation, the worker's expertise and problem-solving abilities can provide quick solutions with minimal or no computational effort. In order to achieve this kind of collaboration, it is necessary to organize the independent tasks of cobots and humans in such a way that when the robot is activated by a particular brain signal, the worker's help is given priority, and a supplementary task is carried out. The behavior shown in Figure 4.12a is that the robot and the worker carry out their respective tasks in parallel until the operator requires assistance, pauses or delays the robot's primary routine. After the assistance is completed, the robot resumes its main job, and the worker resumes his own. On the contrary, as shown in Figure 4.12b, the cobot may require human assistance if it reaches a point where it cannot continue its task. The operator's problem solving abilities can easily resolve a difficult situation by supplying an input signal to the robot, thus making the programming of the cobot task less complex. Once the problem has been solved, both parties can go back to their primary duties. In this task planning strategy, the individual and independent tasks of the two actors can be combined with collaboration on demand. An intermediary is necessary to link the BCI headset to the robot controller, and the Robot Operating System (ROS) is the ideal choice for this purpose. An example of a multitask routine is depicted in Figure 4.13, which consists of three main blocks: a BCI signal acquisition and processing block, an ROS communication management layer, and a TM controller logic block.



(a) *(b)*

Figure 4.12: a) The robot assists the human operator when triggered by him b) If the robot triggers an interrupt the human operator helps the robot with a simple input command

At each cycle, ROS inquires the BCI block about the frequencies that have been read. If there is no match to the frequencies used to stimulate SSVEP, the robot is given priority and its primary task is carried out if no interaction with the operator has been triggered.
If an exception is identified in the TMflow block while the robot is performing its task, the operator is requested to provide assistance by selecting the appropriate action / direction with the BCI device. Robot task execution is given precedence, but the worker's input is taken into consideration.
When an input is read from the BCI block, but no exception request is received from the robot, it implies that an assistance request has been made by the operator. In this case, the robot prioritizes providing human assistance before continuing with its main task.
If no exceptions or external requests are encountered, the algorithm runs the loop as is.

To provide input to the robot, the worker is requested to watch the SVEEP stimuli corresponding to the action that solves the robot problem, which is programmed in advance. The Steady-State Visually Evoked Potentials (SSVEP) response is a reactive BCI technique, based on electroencephalogram (EEG) signals. When the operator, wearing a headset, is subjected to a visual stimulus, the signals gathered from the response of the visual cortex area, suitably processed and analyzed, allow to provide inputs to the robot controller. In the proposed layout, the EEG signals are acquired using a cap with electrodes located in accordance with the international 10-20 system [105], whose representation is reported in Figure 4.14. SSVEP signals are recognized when the retina is subjected to blinking visual stimuli in the band of $3.5 - 75Hz$ [104] even if, in common practice, the upper bound of the frequency range of the stimuli can be limited to $20Hz$ [102]. SSVEP signals are characterized by a high Signal-to-Noise Ratio (SNR) [103, 102], thus it is possible to recognize different stimuli with low processing effort and good precision. The visual cortex area reflects the same frequency of stimulus.
Two types of commercial electrode can be used for the purpose: wet, with an appropriate electrolyte gel applied to sensors, or dry [106]. Practical use of dry electrodes would be easier in industrial environments, since it is not necessary to apply gel to allow electrical contact.

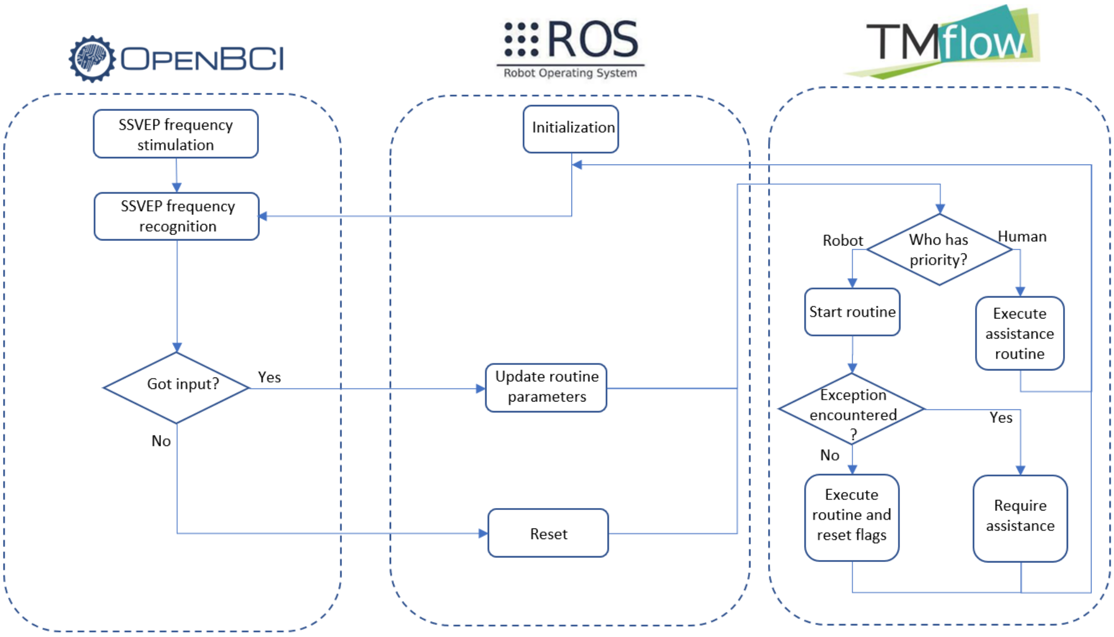


Figure 4.13: Interaction algorithm integrated into the main robot task. OpenBCI is used to acquire brain wave signals, ROS is the communication medium, and TMflow is the robot icon based software

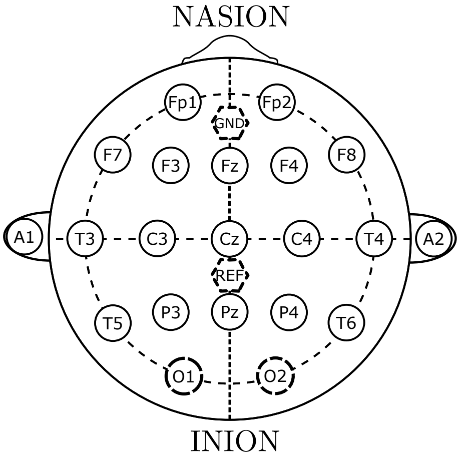


Figure 4.14: International 10-20 system. The subset of exploited electrodes O1, O2, REF and GND is highlighted with dashed bold line.

To implement a suitable methodology for industrial applications, the least number of sensors must be used: according to the 10-20 system, the response of the visual cortex area can be acquired through the O1 and O2 electrodes highlighted in Figure 4.14. Electrodes O1 and O2 are placed in the occipital area, which is responsible for visual processing [107]. Ground electrodes (GND) and reference electrodes (REF), placed in the midline sagittal plane of the skull, must be used to reference and denoize the signal. When the operator is subjected to a visual stimulus blinking at a given frequency, the same frequency and its multiples can be identified in the signal spectra. The generable visual stimuli depend on the type of devices adopted. In this work an LCD monitor with a refresh rate of $60Hz$ is used, in which blinking windows are displayed [108, 109], corresponding to command messages for the robotic application. An OpenBCI headset and a Cython-Daisy board are used in the setup, operating at a sampling rate of $125Hz$. The OpenBCI GUI software receives signals from the Cython-Daisy board, over the LSL communication protocol.

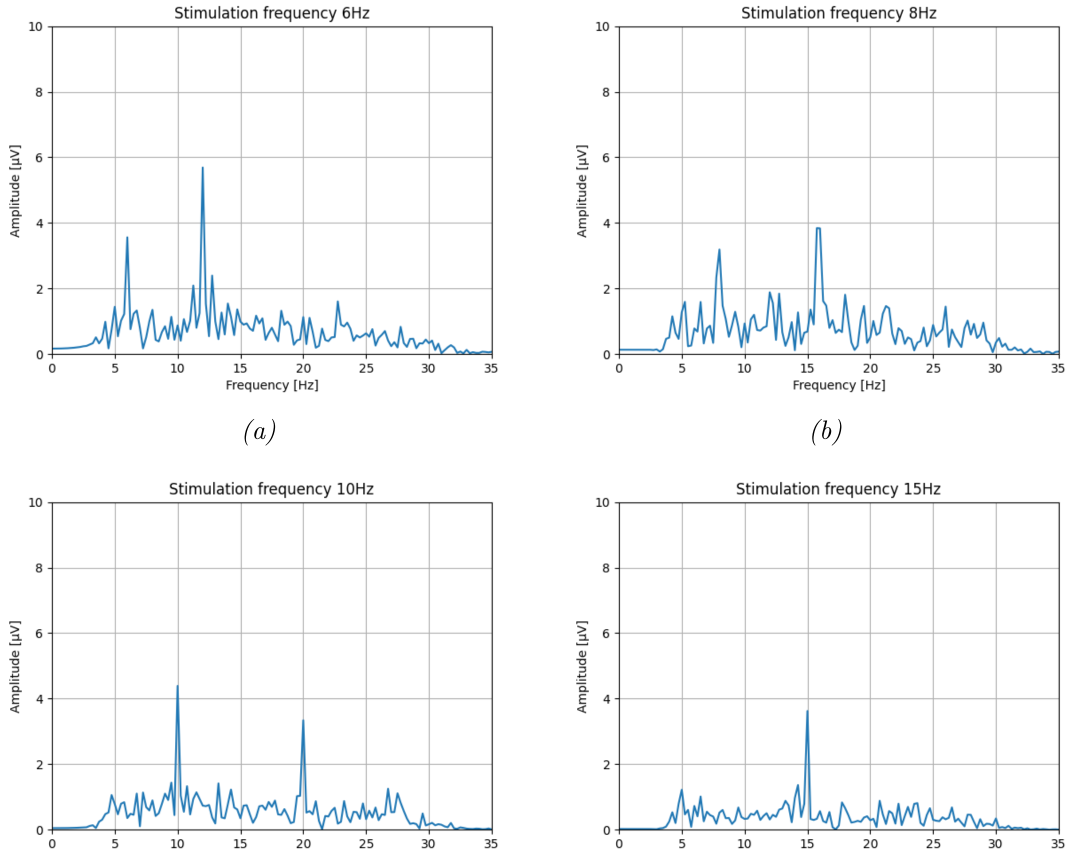
In this section, two experimental scenarios were considered to exemplify the collaborative interaction between cobots and human operators. In the first scenario, a human operator, occupied with tasks that require manual dexterity, employs a Brain-Computer Interface (BCI) to communicate with the cobot. This interface allows the operator to command the cobot to grasp objects or provide assistance, with actions triggered by brainwave signals and specific commands conveyed via SSVEP-based BCI recognition. The second scenario addresses object recognition using the cobot's 2D camera, with potential out-of-view errors mitigated by human-directed repositioning via the BCI. The operator specifies the search direction through SSVEP stimulation, guiding the cobot until the target is found, fostering seamless collaboration.

4.3.2 Experimental BCI communication

In order to evaluate the proposed solutions, a collaborative Techman TM5-700 robotic manipulator is used, along with an Electrode Cap and the OpenBCI CytonDaisy 16 channel biosensing board, for the acquisition of brain signals, described in sections 3.1.1 and 3.3.3, respectively. The robot's primary task is programmed using TMflow software, a user-friendly icon-based interface with integrated visual servoing blocks. Additionally, the TM robot is equipped with an eye-in-hand 2D built-in camera.
Brain signals are acquired using the OpenBCI GUI to view, record, and stream data from the OpenBCI board. The signal is streamed through the Lab Streaming Layer (LSL) and processed by a Python code. The output of this processing is an integer that is used to modify the robot's routine before it is executed. This input is sent to the robot through the TCP/IP connection using the Robot Operating System (ROS) framework.
To distinguish the triggering brain signal, the considered method is steady-state visual evoked potential (SSVEP) signal analysis, with an LCD video terminal providing visual stimulation. According to [110], the time segment factor has a strong influence on the accuracy which can be increased with a time segment longer than 2s, corresponding to a mean accuracy of 90.33%, while a time segment length greater than 3s is enough to obtain recognition higher than 95%. The gain of performance over 4s is not very significant, as the gain of accuracy does not seem worth the time segment extension. For the experimental setup, the 4s segment is considered in the first instance. The FFT analysis is performed to find the frequency at which the maximum amplitude is detected. This frequency is then compared with the stimulation frequencies provided by a $60Hz$ LCD monitor.

The graph in Figure 4.15 displays the FFT analysis of the BCI readings in relation to different stimulation frequencies. The amplitude of the signal is shown in μV on the y-axis, and the frequency spectrum is shown in Hz on the x-axis. As discussed in [102], lower SSVEP stimulus frequencies are found to induce a response spectrum that contains multiple harmonics. This happens for stimulation at $6Hz$ (Fig. 4.15a) and $8Hz$ (Fig.4.15b). For these cases, the local ”peak frequency” is considered with respect to the maximum amplitude, together with its multiples, namely $12Hz$ for stimulation at $6Hz$ and $16Hz$ for stimulation at $8Hz$. The set $6; 8; 10; 15Hz$ is well suited for our purpose, since the SSVEP responses are recognizable, as can be seen in the figures 4.15a, 4.15b, 4.15c and 4.15d.

Once the stimulation matches the response of the visual cortex of the subject, the value of a parameter of the robot routine is updated, as well as the robot behavior, following the flow chart of Fig. 4.13. To communicate with the TM controller, an ROS message is published on the topic to which the TMdriver listening node is subscribed. The ROS environment runs the standard ”roscore” and the ”tm_driver” node which represents the robot. To these, a ”brain” node is added to acquire the SSVEP analysed signal and a ”set_parameter” node is required to send the command line to the robot in order to update the global variable linked to the routine or to directly send a move command.



(a)

(b)

(c)

(d)

Figure 4.15: a) Stimulation SSVEP frequency at 6Hz triggers a response at 6 and 12 Hz b) Stimulation SSVEP frequency at 8Hz triggers a response at 8 and 16 Hz c) Stimulation SSVEP frequency at 10Hz triggers a response at 10 and 20 Hz d) Stimulation SSVEP frequency at 15Hz triggers a response at 15 Hz

A multitask collaborative workstation can be created with the proposed algorithm, allowing the user to collaborate with a cobot using a BCI device. This was tested in an experimental setup that included OpenBCI GUI, Python code, and ROS workframe, which are open-source tools. It was discovered that by activating an ROS communication node and equipping the robot with a set of basic task-based functionalities, human-robot communication through brainwave signals can be simplified, allowing flexible hands-free control of the workstation.

4.4 Brain–Computer Interface and Hand-Guiding Control in a Human–Robot Collaborative Assembly Task

In this section focuses on a specific collaborative assembly task. A brain computer interface (BCI) is exploited to supply commands to the cobot, to allow the operator the possibility to switch, with the desired timing, between independent and cooperative modality of assistance. The two kinds of control can be activated based on the brain commands gathered when the operator looks at two blinking screens corresponding to different commands, so that the operators do not need to have his hands free to give command messages to the cobot, and the assembly process can be speeded up. The feasibility of the proposed approach is validated by developing and testing the interaction in an assembly application. Cycle times for the same assembly task, carried out with and without cobot support, are compared in terms of average times, variability, and learning trends. Therefore, the usability and effectiveness of the proposed interaction strategy are evaluated to assess the advantages of the proposed solution in an industrial environment.

In this section, we test the concept of collaboration on demand by setting up a Brain Computer Interface (BCI) to transfer command messages from the operator to the cobot, thus enabling a supportive behavior in which the operator can use hand guiding control to be assisted by the robot. The communication strategy and the collaboration on demand is deployed in a proof-of-concept assembly task developed on a TM5-700 cobot. The BCI allows the operators to keep their hands free for the assembly task. Moreover, the possibility to insert few BCI sensors in personal protective equipment (i.e. helmets) would pave the way for application in industrial environments.

4.4.1 Collaboration On-Demand Strategy exploiting BCI and Hand Guiding

The different phases of an assembly process can be classified based on their repetitivity and complexity. Robots can handle the most repetitive and dull operations, while humans can execute complex activities. Collaborative robots in assembly tasks can also provide benefits in handling large and heavy objects [111], thanks to the hand-guided operational mode and the ability to reduce the apparent mass of heavy work pieces by a factor of ten or more [112]. This means that the physical strain of workers is significantly reduced.

The cooperation between human and cobot depends on the synchronization between the different phases and on proper task assignment. In a collaboration-on-demand strategy, the workcell is human-centered, and the robot is intended as a supportive tool for the worker. Figure 4.16 schematically represents the process, highlighting the alternating switch between two different phases, namely *independent* and *supportive* phases. During the *independent* phases, the robot and the operator work in a co-existence scenario, operating on different workpieces and processes. As soon as the operator needs the robot assistance, the supportive phase can be enabled: the operator and robot work on the same workpiece interactively, and the robot must behave according to the human intentions. This requires the robot to be informed of human intentions in advance, through a command message [24] that the worker can send through a suitable user interface in the desired time. The operator can therefore use the robot as a flexible tool whose supportive behavior is triggered at will, according to his needs and timing. When the supportive function ends, the independent behavior can start again, and the robot and operator continue to perform their planned activities. In addition, the reverse transition from supportive to independent phase can be controlled by the human through another command message, so that only two different messages are needed to deploy the described interaction scheme.

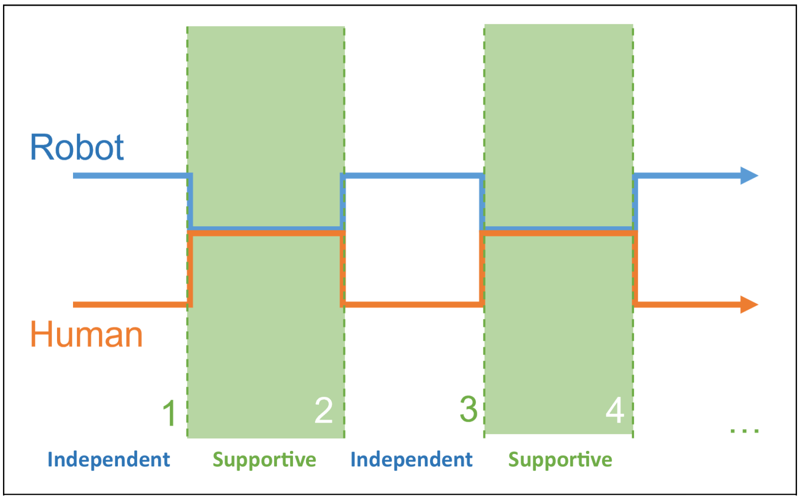


Figure 4.16: Collaboration On-Demand.

In the particular industrial case proposed in this section, the switch between independent and supportive phases is controlled through a BCI, which allows us to send a command message without the use of hands. BCI interface is used in reactive mode with the Steady-State Visually Evoked Potentials (SSVEP) method [101]: the operator looks at an external monitor with images blinking at two different frequencies corresponding to the two command messages needed to switch between the two operating modes. During the supportive phase, hand-guiding control is exploited to position the objects in a co-manipulation mode. Hand guiding control has been realized by means of a load cell mounted on the robot wrist.
An exemplary assembly task has been considered as a test case consisting of pick-and-place operations, relative positioning of objects, and joint connections with bolts and nuts. The task can be, therefore, divided into simple, repetitive operations like pick-and-place and more complex ones like joint connections. The former are assigned to the robot, whereas more complex manipulations subtask can be left to the operator. As soon as the cobot has picked up the assigned component, it positions itself in a stand-by pose, waiting for the operator to take control of the process. As soon as ready, the operator switches to supportive operational mode, so that the intermediate step of positioning the item relative to the previous one can be done in a cooperation mode, through hand-guiding control. The robot can

then assist the operator by holding and moving large and heavy objects according to its payload, leaving the operator the flexibility to properly position the component with the desired timing and then to join it to the other parts of the structure being assembled.

Figure 4.17 shows the proposed framework in which the operator can interact with the robot manipulator through two different interfaces: the activities related to the BCI are described on the left side of the figure, whereas those related to the load cell and the haptic control are reported on the right side of the scheme. On the left-hand side of the scheme, the SSVEP signals are collected by the electrodes, processed, and then referenced to suitable *command messages* to be sent to the robot controller. On the right side of the scheme, the load cell provides a haptic interface with a hand-guiding control, which introduces *guide messages* to properly position the robot. The task flow depends at the same time on the preprogrammed robotic subtasks and on the real-time commands given by the human operator. The latter makes the decision on when the robot must move independently and when to switch to a supportive phase.

Each BCI-SSVEP signal is assigned to a predefined specific robotic function. When the operator and the robot end their independent phases, the operator can activate the hand-guiding mode providing the command 'Haptic On'. The command 'Hold on' is given when the operator needs the robot to keep precisely the same position it has been placed through hand-guiding control, so that the assembly operation can take place. The same SSVEP signal can be used for both cases. When the operator finishes connecting the two parts by screws and bolts, the 'Next' command allows the robot to return to its independent phase. Therefore, the logic applied requires only two signals, hence two different stimuli: one associated with the command message "H" (which can have two different states: "Haptic On" or "Hold On"), and another for the command "N" ("Next").

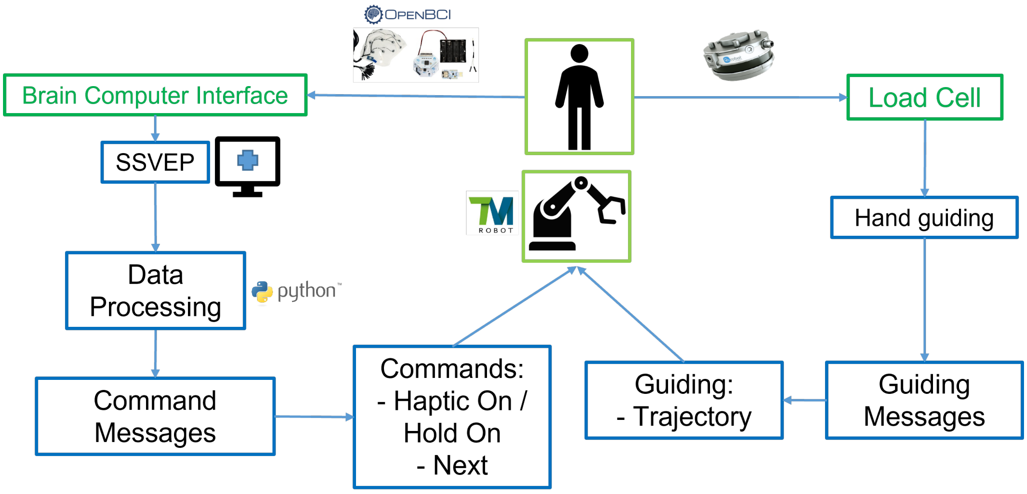


Figure 4.17: BCI-SSVEP with Load Cell sensor Framework.

Figure 4.18 represents the same on-demand collaboration framework from the perspective of the human operator and the robot, which are seen as parallel players in the task. Command and guidance messages are in charge of timing and controlling the task: when the operator sends the "Haptic On" command, the two branches of the independent phases merge, starting the supportive phase, in which operations are headed by the operator and supported by the robot. After the hand guided phase, in which the operator can place the component hold by the robot relative to the structure being assembled, the "Hold On" command allows to keep the component in place, so that bolting operations can take place. Afterwards, the "Next" command makes the operator and robot get back to their independent phases: the former can continue to perform assembly operations for which he does not need any support, the latter can pick the next component to be handed to the operator. He will pick it up just when he needs it, and he will take charge of the process through another "Haptic On" command.

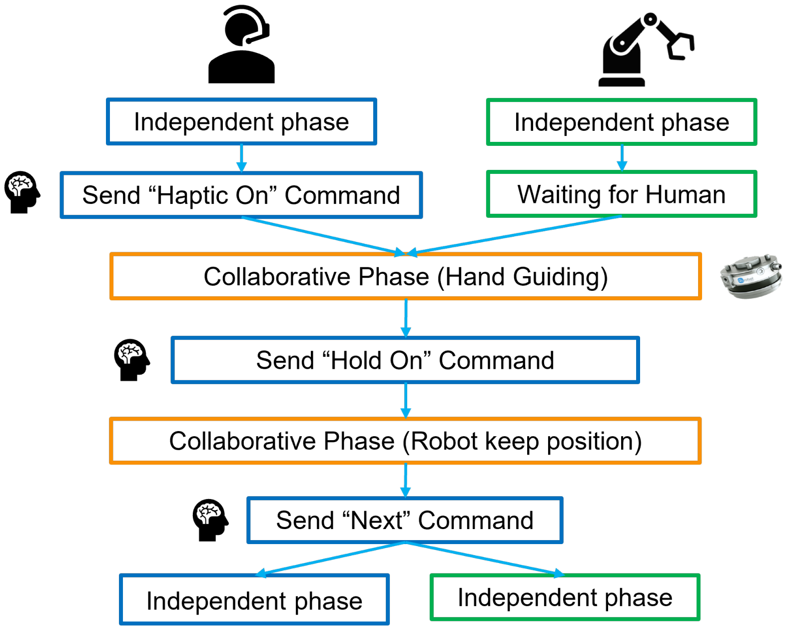


Figure 4.18: Collaboration On-Demand with BCI and Hand Guiding.

4.4.2 Brain Computer Interface based on SSVEP analysis

The SSVEP frequency recognition software and its integration with the robotic workflow have been specifically developed. The algorithm flow diagram is shown in Figure 4.19: when the robotic task starts, the first step of the SSVEP frequency recognition program is to open communication with the OpenBCI software and to flush the data buffer. Then the signals O1 and O2 of the visual cortex area are gathered with a time window of $1s$. They are averaged and filtered using a digital 4th-order Butterworth bandpass filter in the range $4Hz - 45Hz$. Finally, a Hamming window is applied to the signal, and a Fast Fourier Transform (FFT) is performed to identify the frequencies related to the visual stimulus. The FFT main peaks are analyzed to seek for peaks that exceed a certain threshold at pre-set frequencies (the stimulus frequency and its first two multiples), which correspond to an actual will of the operator to give a command message.

These operations are performed in a continuous loop, and the corresponding command messages are sent to the robot only if a "cooperative flag" has been activated, meaning that the robot has completed its independent phase and it is available to shift to cooperative operational mode. This prevents false positive commands from being transmitted to the robot before its independent phase has been completed.
The robot commands corresponding to each command message are programmed in the robot controller and triggered by the PC when the corresponding command message is detected. Signal processing software, as well as the one in charge of sending commands to the robot, has been implemented in Python 3 with the SciPy and Pymodbus libraries.

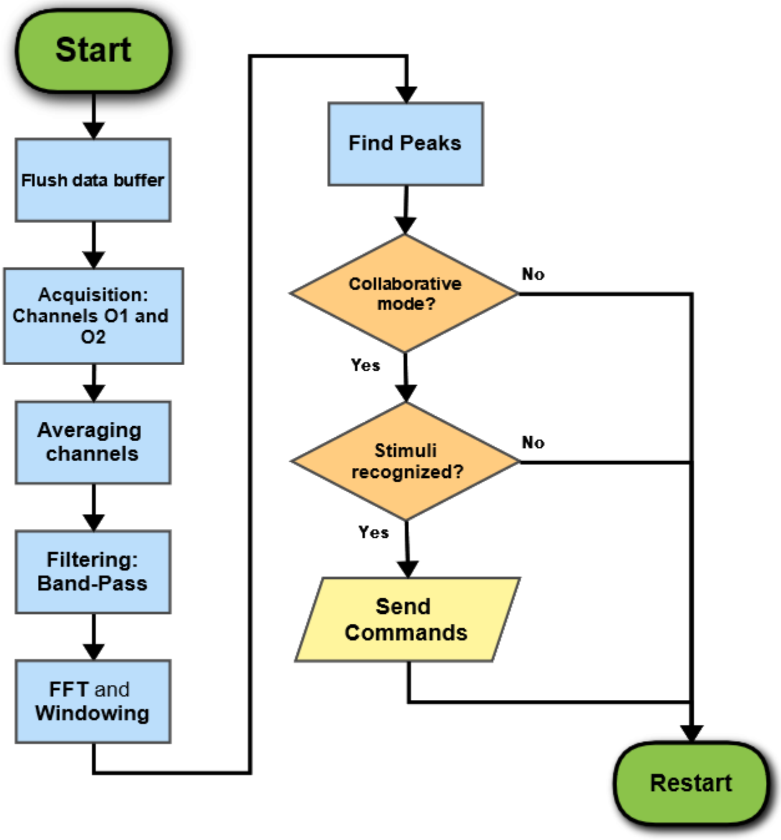
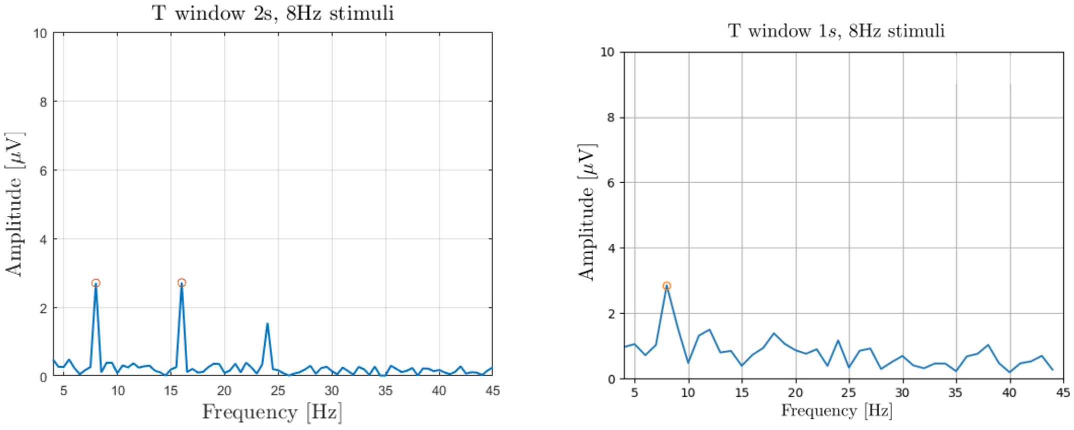


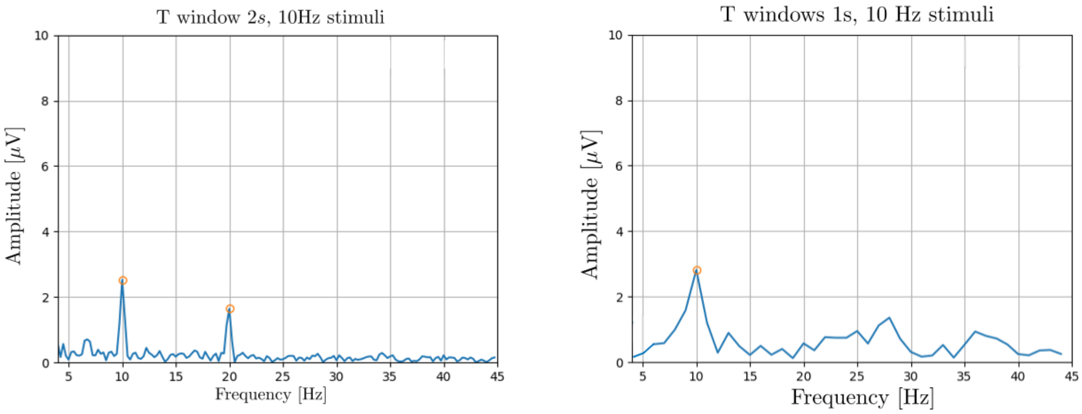
Figure 4.19: SSVEP frequency recognition software flowchart.

The issue of having a quick reactive response is essential for the fluidity of the robotic task. For this reason, to improve the speed of the system, a time window length of $1s$ has been tested for frequency recognition in SSVEP and compared to one of $2s$. Figures 4.20 and 4.21 report the SSVEP responses to a visual stimulus blinking at $8Hz$ and $10Hz$, respectively. For each frequency, the results corresponding to the time windows of $2s$ and $1s$ are reported in Figure **(a)** and Figure **(b)** respectively. For all the considered cases, it is possible to detect the main frequency of the stimulus, with the first multiple frequency also visible in the case of higher spectral resolution. The challenge of motion artifacts had to be addressed as the operator was in motion during the assembly task. Movement artifacts lead to a signal spectrum with multiple peaks scattered throughout the examined frequency range, as described in [113]. This could lead to the creation of erroneous and unintended orders. This problem had to be addressed during the signal processing stage by eliminating all occurrences that had a peak at frequencies that did not match the anticipated response.



(a) *(b)*

Figure 4.20: FFT analysis of SSVEP response for a $8Hz$ *visual stimulus.* ***(a)*** *Time window* $2s$*.* ***(b)*** *Time window* $1s$*.*



(a) *(b)*

Figure 4.21: FFT analysis of SSVEP response for a $10Hz$ *visual stimulus.* ***(a)*** *Time window* $2s$*.* ***(b)*** *Time window* $1s$*.*

4.4.3 Hand Guiding Control

In the proposed assembly task, positioning and comanipulation operations require guidance messages for motion control, and in particular for the proper positioning of the workpiece to be joined to the structure being assembled. Hand-guiding control is suitable for this purpose, assisting the operator in positioning and sustaining the weight of the parts to be assembled. In the experimental setup, hand-guiding control has been developed exploiting an *OnRobot HEX-E* force sensor introduced in 3.3.1, measuring 6 axis components, mounted on the robot wrist. The system is able to sense the gripped workpiece and compensate for its weight. The arm is guided according to the forces sensed and imposed by the operator's hand, based on the proposal explored by [111, 114, 115], with a control scheme adjusted to fit the programming features of the *Techman TM5-700* cobot exploited. A block *force control* in the TMFlow software, activated during the cooperative phases, allows one to follow any driving force provided by the operator (that is, the force in the x, y, and z directions). According to the ongoing phase in the task and in particular to the piece to be assembled in the predefined sequence, different directions for control can be enabled or disabled, so as to have either a planar or 3D motion. For the first part to be assembled, positioning requires only a Cartesian linear motion in X-Y-Z coordinates. For the second positioning phase, the angular orientation along the Z axis of the wrist reference frame has also been involved. Since only the rotation along the Z axis is enabled, the gravity force always acts along the Z axis of the robot wrist. The force control block integrates a PID control on the sensed forces and torque applied to the end effector. To compensate for the gravity effects on the raised objects, measurements are reset at the beginning of each force control routine, and then stored to be used as reference force/torque for the PID controller. PID gains used for the Cartesian linear motion are: $k_p = 0.15$, $k_d = 0$ and $k_i = 0.00001$; for the rotation along the Z axis: $k_p = 0.012$ and $k_i = k_d = 0$.

4.4.4 Assembly task

The discussed control structure has been validated using an exemplary assembly task developed with a TM5-700. However, the same control structure can be applied to a larger robot, in which the higher payload admitted allows the assembly of heavier components, thus enhancing the effectiveness of the proposed solution in helping the human operator. The task is divided into the following steps:

1. The operator is getting ready to work on a lengthy aluminum profile by constructing a corner joint with screws and nuts, while the robot is selecting a short aluminum profile and placing it in front of the operator, ready to be manually directed. During this phase, the operator and the cobot work with independent collaboration strategies.

2. To activate the cooperative mode through the BCI signal ("Haptic On" command), the operator looks at the monitor window with the letter "H" blinking at a frequency $f_1 = 8Hz$. He can then exploit the hand-guiding control and place the beam in the final position where it is to be assembled. After a second command message is given through BCI ("Hold On" command, blinking letter H), the robot keeps the component in position and the operator can connect the two aluminum profiles with a corner joint and nuts. In this steps all activities are carried out with supportive strategies.

3. After finishing the assembly of the two parts, the operator gives the "Next" command by looking at the "N" letter blinking on the monitor at a frequency $f_2 = 10Hz$. The operator prepares independently a second corner joint and gathers bolts and nuts, and the cobot executes the next routine, picking another

long aluminum profile, handing it to the operator, and waiting motionless for the next BCI message. In this phase, the operator and robot work independently.

4. When ready, the operator activates once again the cooperative mode through the BCI signal (”Haptic On” command) and hand-guides the beam into the proper position to be assembled. After a second command message through BCI (”Hold On”), the robot keeps the component in position, and the operator can join the two aluminum profiles with a corner joint and nuts. Once assembly is complete, he/she communicates to the robot the end of the task, through the last ”Next” command.

The following figures from 4.22 to 4.24, highlight the main steps of the described assembly task. Figure 4.22 shows an independent phase, representative of phases number 1 and 3 in the numbered list above, where the operator mounts an angular joint while the robot places the beam. It is possible to notice in the background the monitor with the blinking windows that are used as visual stimuli for BCI.

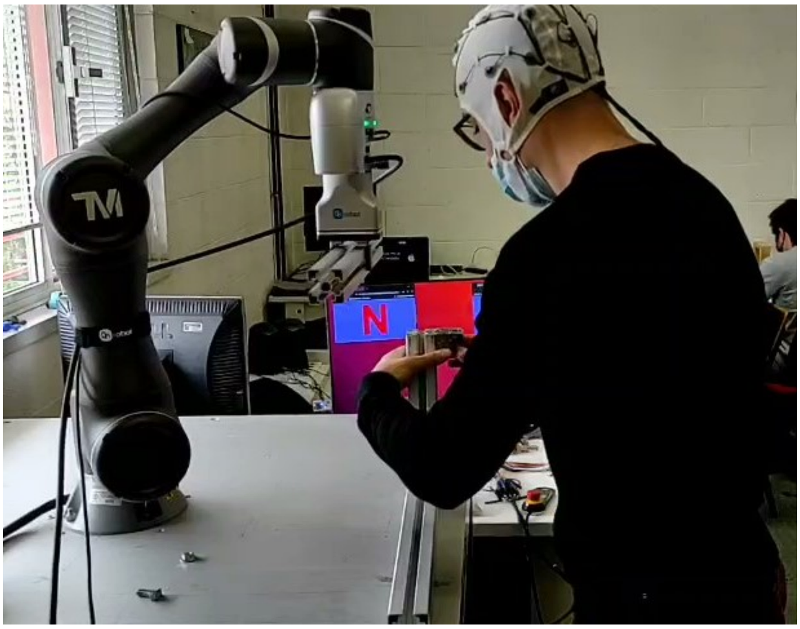


Figure 4.22: Independent phase. Step 3 of the assembly task.

In Figure 4.23 the switch from independent to supportive phase is enabled by the operator who looks at the “H” letter on the monitor, to activate the hand guiding mode. This switch only takes 1 second, due to the FFT analysis discussed in the previous section 4.3.1.

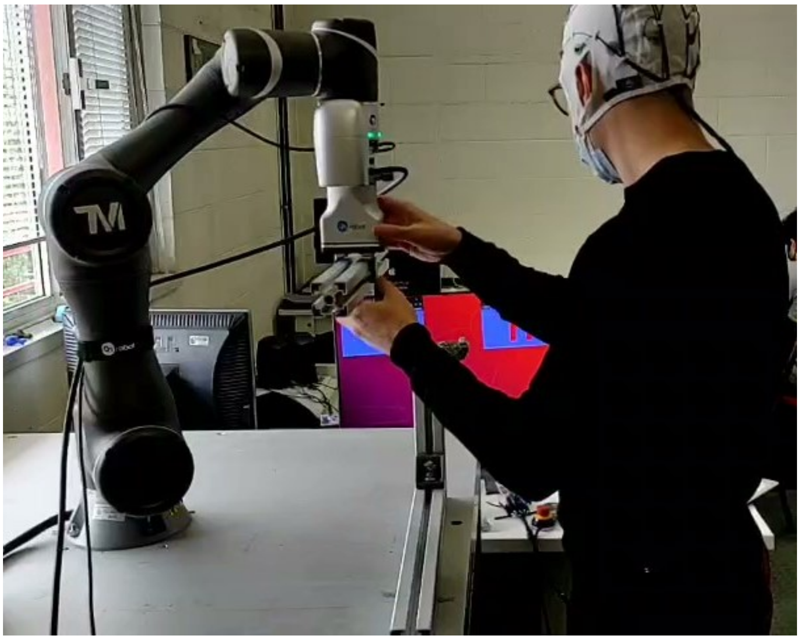


Figure 4.23: The operator looking at "H" window is activating the hand guiding.

Finally, Figure 4.24 shows the supportive phase in which the operator exploits hand guiding control to position the beam to be assembled. When a proper positioning has been achieved, the hand-guiding control can be deactivated looking at the "H" window again, so that the robot holds the component steady in the desired position. Then, when the operator finishes the assembly, he can move the robot looking to the "N" window. In this way, the operator gives the proper task timing so that the robotic code can obviate any difficulties or setbacks in the assembly phase.

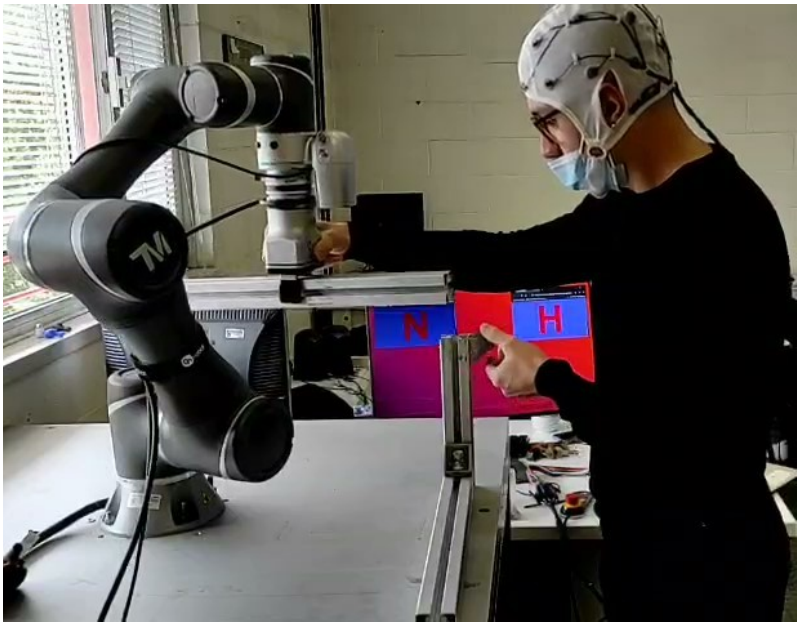


Figure 4.24: Step 4 of the assembly task. Supportive phase with hand guiding.

Figure 4.25 shows the flow chart of the robotic program. Independent phases are represented with yellow blocks, decisional phases in which the robot waits for the input of the BCI are colored blue, and supportive phases are colored green. At the beginning of the program, initialization activities and variable assignments occur. Subsequently, the subtasks to be executed in independent mode are executed (e.g., picking up the bar, moving to the handing pose to wait for the operator).Once the independent phase is finished, the cobot is ready to enter the cooperative mode. It turns on the ”cooperative flag”, which allows it to receive the BCI command ”H” from the operator. Then it waits for the operator's command: measurements from BCI device and results from data analysis start being taken into account by the robot controller. SSVEP frequencies gathered from the BCI headset are compared with the pre-set frequencies to recognize the operator's choice. The program moves forward when the frequency corresponding to the H signal is detected. The H command can have two binary states. In the former state, the hand-guiding control is activated. In the latter, the hand guiding is deactivated, and the system goes into a new decision block, waiting for the next (N) BCI command. At this point, the second independent subtask starts, and the described cycle is repeated.

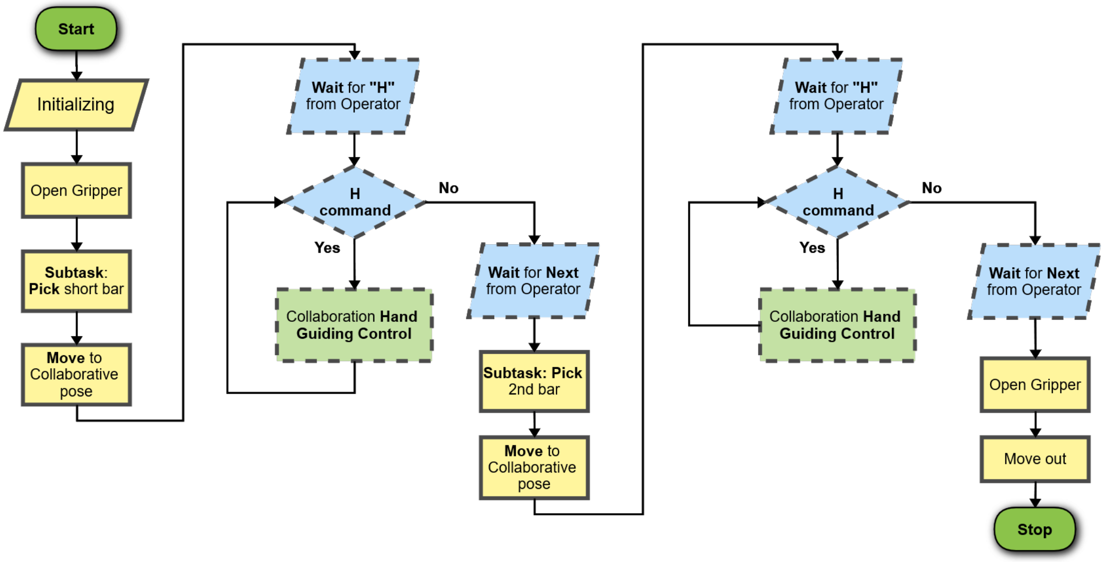


Figure 4.25: The robotic task flowchart.

There is a possibility that problems may occur due to incorrect or inadequate signals from the interfaces during the task. These are associated with the Brain-Computer Interface or the manual guidance control signals.
Regarding command messages (i.e. through BCI), the following cases can be highlighted:

- A low signal-to-noise ratio, caused by a poor quality of the electrode/gel and skin contact, can cause a non-acquisition of the frequencies associated with command messages, leading to a slowdown of the task as the robot will keep on waiting for a command through BCI.

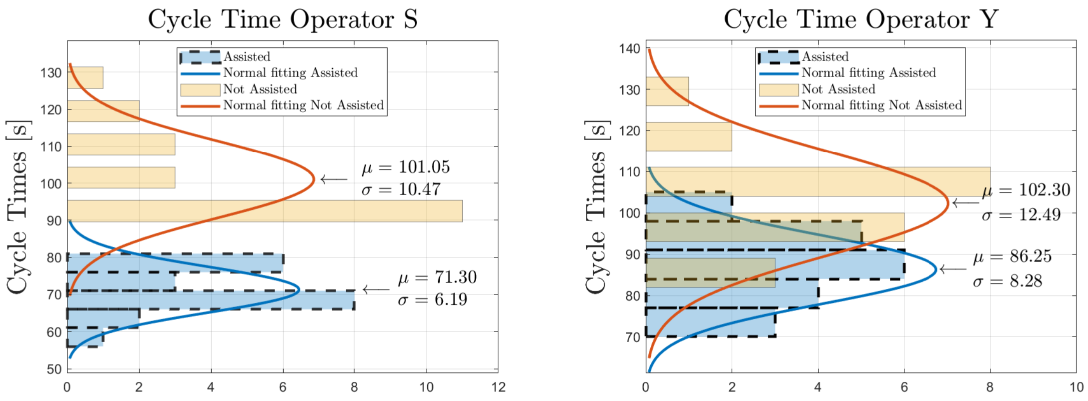
- In the presence of motion artifacts, the commands are ignored. This is a positive feature if the motion artifact is to be discarded, but if motion artifacts occur right when the operator is turning his head to look at the blinking signal, the desired command message would not be delivered. A new signal acquisition would be required, causing a delay of a few seconds in the overall task.

- The peripheral vision of the operators might lead to the acquisition of an incorrect command during the task, even if the operator is not purposely looking at the blinking signals. The influence of peripheral vision can be reduced by locating the blinking stimuli far from the peripheral field of view.

If the robot is guided by the operator in a singularity position, an incorrect guidance message may be produced. Since the hand-guiding control uses the motion control operating in the working space, in singularity positions the motion control fails to compute the inverse kinematic, leading to a failure of the collaborative mode. This possibility can be completely avoided and can be considered human error.

4.4.5 Experimental results

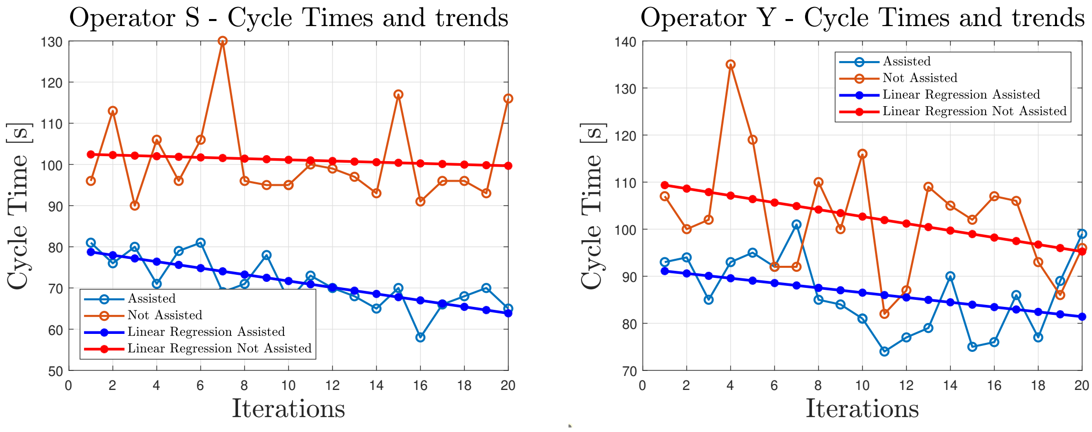
In order to assess the potential improvements achievable through the collaborative strategy described, two series of tests have been carried out, repeating the assembly twenty times, first purely manually and then with robotic assistance. Both assembly series have been repeated by two different operators (named in the following S and Y) to also take into account variability due to the human factor. The cycle times have been measured for each repetition.
Figures 4.26a and 4.26b show the distributions of the cycle times corresponding to the two operators, S and Y, respectively. In each figure, the cycle times related to manual assembly (labeled *Not assisted*), and to human / robot assembly (labeled *Assisted*) are reported, together with the values of average μ and standard deviation σ and the representation of the corresponding Gaussian distribution. Both operators experienced a clear reduction in average cycle times in the assisted case compared to the unaided one ($71.3s$ against $101.05s$ for operator S and $86.25s$ against $102.3s$ for operator Y, corresponding to a reduction of 29.44% and 15.69%, respectively), mainly due to the assistance of the robot to properly position the component to be assembled by means of the hand-guiding control and hold it in the proper position during bolting. Moreover, the cases in which the operator is assisted by the cobot show a lower standard deviation of the cycle times, meaning that the presence of a predetermined workflow mitigates the variability of the cycle times, helping the operator to be more regular in his operations. The standard deviation of the cycle times is, in fact, equal to $6.19s$ against $10.47s$ for the operator S, $8.28s$ against $12.49s$ for the operator Y.



(a) *(b)*

Figure 4.26: Assembly Cycle times distribution for operators S and Y.

The cycle times of the test series are shown in Figure 4.27 in the order in which they were collected. These are the same data as in Figure 4.26. A decrease in cycle time is evident in the test series as the number of repetitions of the task increases. This is due to operators honing their skills as they become more familiar with the activity. The linear regression lines plotted in the figures can be used to estimate the learning rate of each operator. The linear regression slopes in the assisted cases show a reduction in cycle time for both operators ($0.78s/iteration$ for operator S, $0.51s/iteration$ for operator Y), highlighting a beneficial effect of the robotic system. On the other hand, in the case without assistance, a more erratic behavior is observed, with operator S showing almost no learning ($0.15s/iteration$), and operator Y showing a relevant learning rate ($0.74s/iteration$). The results are summarized in Table 4.1.



(a) *(b)*

Figure 4.27: Learning rates. (a) Operator S. (b) Operator Y.

Table 4.1: Results summary table: Assisted vs. Not Assisted

Assisted vs. Not assisted cases summary table				
Operator		Avg. Cycle time μ	Cycle time σ_{STD}	Learning rate
Operator S	Not assisted	101.05 s	10.47 s^2	0.15 s/*iteration*
	Assisted	71.30 s	6.19 s^2	0.78 s/*iteration*
	Percentage difference	−29.44%	−40.88%	80.77%
Operator Y	Not assisted	102.30 s	12.49 s^2	0.74 s/*iteration*
	Assisted	86.25 s	8.28 s^2	0.51 s/*iteration*
	Percentage difference	−15.69%	−33.71%	−31.08%

4.4.6 Discussion and limitations

A collaboration on-demand strategy is implemented in the present section, with the aim of demonstrating the possibility to exploit BCI interface to give command messages to the robot controller in an industrial assembly task. The BCI interface provides the operator the chance of controlling and giving proper timing to the robotic task during the assembly operations, without the need to use hands to push physical buttons or to interact with a gesture recognition system. A proper BCI command can switch the robot operating mode from independent to cooperative, thus activating a hand-guiding control by which the cobot and the human operator can interact in a supportive manner during the assembly task. The performance of this approach has been experimentally validated with two different operators, resulting in a significant reduction not only of the average cycle time (−29.44% and −15.69%), but also of its variability (standard deviation of the cycle times), thus leading to a more predictable productivity.

Currently, BCI technology is not commonly used in robotics for industrial purposes, such as in assembly operations. This research suggests that this technology could be a viable option for Industry 5.0 applications. Despite the advantages of shared tasks, certain restrictions remain. To ensure successful completion of the task, the entire assembly process must be carefully planned in advance, taking into account the different stages that will be handled by the human and the robot. This requires a rigorous design phase. In comparison to traditional robotic applications, the timing control of the assembly task is in the hands of the human operator, shifting the control of the task to a more human-centered approach. This can also lead to delays caused by the operator's behavior, due to human errors, but also to the technological restrictions of the chosen interface. The signal-to-noise ratio of the data generated by the BCI, which is usually quite high, can vary depending on the person wearing the BCI helmet. In such cases, the peak of the expected frequency may take more than one time window to be identified, which can have a negative effect on the total execution time.

This work addressed motion artifacts by discarding any occurrences that had peaks at frequencies that did not match the expected response. In this case, the command would not be sent, resulting in a delay as a new signal would need to be acquired. The issue of signal-to-noise ratio, depending on the individual wearing the helmet, and motion artifact are still topics that require further exploration in the progression of this research.

4.5 Human-robot collaborative assembly assessment

Generally, working environments designed for humans and conventional robots do not exhibit overlaps, primarily due to the fact that the latter operate within cages or protective enclosures. This separation arises from the absence of the necessary safety requirements in automated industrial robots to allow close proximity with humans. However, within the realm of Human-Robot Interaction (HRI), tasks performed by both the human operator and the robot are integrated within a shared workspace, thus breaking down the traditional barrier between manual and automated work.
According to [23], assembly tasks are typical tasks that are suitable for collaborative robotic applications, since the action of the operator influences the behavior of the robot and vice versa. However, the most common use of collaborative robots for assembly tasks on manufacturing lines consists of workstations suited for sequential assembly [47], in which the robot performs the simpler operations, and the more complex or variable ones are left to the human. Whenever possible, the worker performs the last manipulations on the assembled product at the end of the assembly line to limit the need for interaction with the robot. Different collaborative strategies can be used according to the degree of interconnection and the dependency of tasks [27].
A more cooperative parallel assembly is characterized by a human intervention that takes place in parallel to robot activities, so that the last assembly steps can also be carried out by the robot. In this second scenario, timing and coordination between human and robot are critical factors that might severely affect collaboration, which therefore strengthens the need for a suitable interaction strategy and task planning.
In this section, an assembly task is taken into account as an exemplary process for collaborative robot integration into an originally manual human activity. The task is similar to that described in the previous section 4.4. It consists of mounting a structure made of aluminum bars and angular joints as depicted in Figure 4.28. The first bar is grounded to the table onto which the task is performed (Fig. 4.28-1). The second bar is attached to the first at an angle of 45 degrees by fixing an angular joint between the two (Fig. 4.28-2). The last bar is mounted perpendicularly to the second (Fig. 4.28-4), after a second angular joint has been attached to its edge (Fig. 4.28-3).
When the possibility of direct command message from the operator to the robot is enabled through a suitable user interface, human intentions can be communicated to the robot. Then it is possible to alternate between the *independent* strategy, in which the operator and the robot work simultaneously and independently on their own tasks, and the *supportive* strategy, in which the operator receives assistance from the robot. The switch between *independent* and *supportive* phases can be enabled according to the time given by the operator, who independently works on his own task until he gives a triggering signal to the robot. The "Collaboration On-Demand" strategy, discussed in section 4.3.1, allows the operator to switch from independent to supportive phase. Once the machine detects a consent exchange, the operator can be assisted by the robot in a supportive manner. HRI is a key factor that is assuming an increasingly significant role as more robots are being programmed to work alongside humans within industries.

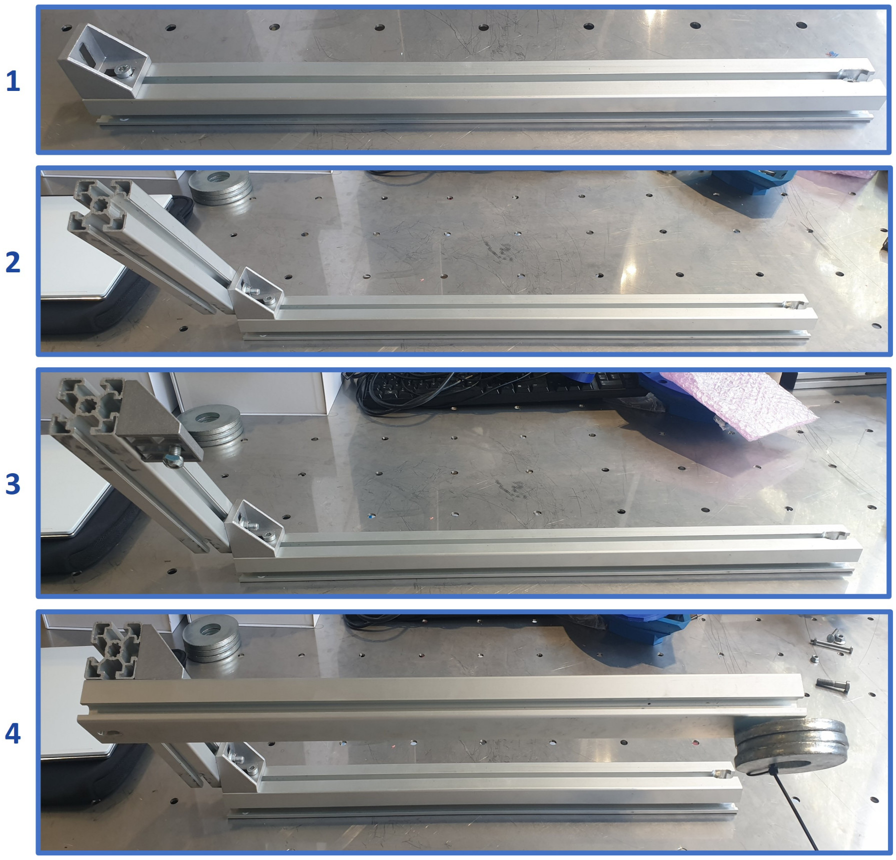


Figure 4.28: Sequence of the assembly sub-task of three bars and two angular joints

4.5.1 Task allocation

Task allocation is a critical element in industrial automation, particularly when it comes to creating a collaborative environment between human workers and cobots. Before allocating any specific functionality to humans or robots, it is necessary to perform a task analysis that involves a thorough examination of the activities involved in a particular production process or task. The aim is to identify which parts of the process can be automated or aided by cobots and which still require only human involvement. This analysis takes into account various factors, such as the complexity of tasks, accuracy requirements, frequency of execution, and, most importantly, safety. A comprehensive understanding of tasks is essential to determine how cobots can be optimally integrated into the work environment. Task allocation is the final process by which specific tasks are assigned to both human operators and cobots within the collaborative environment. This allocation must be done carefully to ensure overall efficiency and, most importantly, the safety of operations.

In the process of task allocation between humans and cobots, two different higher-level methods can be identified. Static (off-line) task allocation techniques determine the task arrangement prior to its actual implementation, meaning that once the assignment of tasks between humans and robots is established, it cannot be altered during execution. The robot must be stopped, reprogrammed, and then the task can be carried out with the updated parameters. The downside of this approach is that it often leads to a tedious working environment, likely resulting in a decreased level of situational awareness and a decreased capacity to learn. These drawbacks are exactly the issues that the human centricity of industry 5.0 is trying to solve. The dynamic (online) task allocation method, on the other hand, enables automatic decisions to be made in real-time. According to [116], dynamic allocation procedures refer to algorithms that can respond to situations and events, adjusting the assignments of tasks accordingly. This would require programming all possible scenarios the robot has to react to, or the system would need to learn a variety of eventualities based on collected data to effectively reconfigure based on the new situation. The drawback of predefined programming or machine learning is that operators, again, are not involved in the decision-making process. The robot's behavior is determined by the supplier, integrator, or programmer in an offline phase, and the algorithm is designed to adjust to the objective parameters. Additionally, it would be very difficult for the operator to understand how and why the system assigns tasks. Transparency and explainability, two increasingly important issues in robotics and artificial intelligence, would thus be difficult or even impossible to achieve. The purpose of this section is to develop a method to transform a manual task into a collaborative one, which involves studying, categorizing, and assigning duties to humans and cobots. In addition, an intuitive, flexible, and transparent task management solution is proposed to enable humans to remain in control of the collaborative system.

The task assignment involves a number of sub-issues that must be addressed before allocating the resource to a part of the process. To be able to determine whether the task should be performed by a human, a robot, or both, the sequence in which the process is carried out must be established. For those processes that are strictly sequential, the order of execution is predetermined and does not need to be taken into account for task assignment. However, for processes that allow parallel execution of subtasks, which are prepared in parallel and eventually integrated into the overall product, the sequence is not completely predetermined.

Task analysis

The Task Analysis (TA), as defined in [117, 118], is a methodology that aims to model tasks by defining their objectives and the necessary activities to achieve them. More precisely, it involves a detailed study of how a particular task is executed within a work environment, taking into account the required tools, task complexity, and intermediate steps. Various methods are available for task analysis, including goal-directed task analysis, cognitive task analysis, and Hierarchical Task Analysis (HTA). The HTA method is a scientific approach, reported in [119], applicable to tasks with a well-defined structure and processes that tend to be executed in a similar way each time. According to HTA, the overall goal of the task is set as the "main goal," and the objectives of various operations (subtasks) that make up the task are defined as "subgoals." Each subtask is then further decomposed into a list of fundamental or functional actions. Thus, according to [120], the output of a complete HTA is a diagram that describes a hierarchy of operations and execution plans. This hierarchical diagram can be represented as a tree structure, allowing for a clear visualization of the functional activities and their connections. According to [119] the HTA is a highly flexible tool and can be applied to describe various types of processes and serve multiple purposes, such as training, organization of work, error analysis and prediction, and allocation of functions between humans and machines. In this section, the HTA methodology is adopted to decompose the assembly operation into a hierarchical assembly diagram, enabling further research and analysis of collaboration between humans and cobots.

Resource matching

Once the task is clear and its structure is revealed, another necessary evaluation is to compare the capabilities of humans and robots with the demands of the assembly subtasks. The output is a process plan that can be executed with the available resources [121]. A simple and straightforward approach involves allocating ”easily programmable tasks” to the robot and the remaining tasks to the human operator, thus allowing fast reprogramming if required. However, relying on humans to perform tasks that are difficult to automate carries the risk of relegating the employee to a supportive position compared to automation, which must be avoided for the successful implementation of Human-Robot Collaboration (HRC), particularly with respect to worker acceptance, as declared by the authors of [122].

Therefore, in [123], methods for a qualitative comparison between pairs of resources have been proposed. To mitigate the subjectivity of a qualitative evaluation, alternative approaches have been proposed in [124]. The proposed algorithm aims to generate a decision tree as a trained classifier. The classes defined by the classifier include: executable by a human (H), by the robot (R), by both (H/R) and collaboratively (H+R). Other approaches, such as in [125], conduct both qualitative and quantitative analyzes. Multi-criteria quantitative classification techniques were proposed in [126] to assess the suitability of resources for the task. The Analytic Network Process (ANP) is employed to quantify the agent’s ability to perform the task, which is divided into sub-assembly operations.

In contrast to a fully automated system where all data are easily accessible, a manual or collaborative system often has missing or imprecise data. Furthermore, the best agent to perform a task is not always clearly defined. For this reason, the authors of [127] have implemented fuzzy logic-based decision making to evaluate tasks based on multiple criteria and assign them to the most suitable agent.

Although a standardized set of task evaluation criteria for Human-Robot Collaboration (HRC) has not yet been defined, the authors of [128] proposed a Human-Robot Activity Allocation (HRAA) methodology to allocate subtasks in collaborative work cells to humans and robots. The allocation criteria include the following:

- Technical Evaluation Index (TEI): considers the robot’s technical limitations and potential technical issues that may arise.
- Safety and Ergonomics Evaluation Index (SEEI): assesses whether an activity can cause physical stress to the operator or could be hazardous to humans and/or the production environment.
- Qualitative Evaluation Index (QEI): takes into account whether an activity requires process improvements in terms of standardization.
- Economic Evaluation Index (EEI): considers the economic value of the task.

Other studies have proposed different criteria depending on the specific collaborative task to automate. In [129] the authors analyzed physical characteristics, the way the component is placed in the assembly station, and the assembly methods involved. [124] considered four task characteristics based mainly on the weight of the assembled part, its movement, precision, and dexterity requirements. The research criteria presented in [126] consider assembly tolerance and weight of the assembled elements as part characteristics, while ergonomics, strength, and support were considered as agent skills. In a method for planning tasks shared between humans and robots, [127] provided one of the most comprehensive lists of criteria, including ergonomics, safety risk, required dexterity, ambient lighting, surface reflection, variability of component supply, cognitive load, execution time, activity sequence, errors, and quality. Evaluation criteria such as cycle time, distance traveled, resource utilization, and safety were considered in [130]. These evaluation criteria vary according to the specific goals of the HRC process, the complexity of tasks, and optimization objectives such as safety, efficiency, and ergonomics. The choice of criteria will depend on the specific situation and the goals of implementing collaboration between humans and robots in the context of assembly.

The main factors influencing assembly task allocation for the robot are closely related to the components to manipulate. In fact, these are related to the technical feasibility of the robot performing the task efficiently. The load capacity, reachability, or availability of the appropriate gripper have the potential to significantly affect the robot's ability to perform tasks such as feeding, handling, and assembling products. These factors are rooted in the technical attributes of either the product or the production process itself, which can present impediments or increased complexity in the use of collaborative robots for assembly and manufacturing operations. The main components features to consider are the following.

- Size: the largest dimension of the workpieces to handle has to match the sizes of commercially available robot manipulators. This is related to the feasibility of transporting objects without obstructing robot movement during operation. Task execution becomes challenging if the maximum piece size exceeds half of the robot's reach radius. In contrast, if a piece is too small, it lacks sufficient surface area for effective grasp by the manipulator.

- Weight: the maximum value is limited by the robot load capacity. Furthermore, a heavy component will increase the robot kinetic energy during manipulation, raising safety risks for the human operator. This makes lightweight component assembly easier to automate.

- Shape: the geometric symmetry and shape of the assembly components influence the ability of a robot to grasp it in the required position and orientation. The suitability of a component shape for robot manipulation is determined by a sufficient surface for a secure hold. Shape-based evaluation depends on the gripper device options available in a specific context.

- Texture: a flexible behavior of a component, depending on its structure and material, may cause failures of the gripping procedure during manipulation.

Such components can be handled more effectively by humans due to the adaptability of human hands.

- Fragility: fragile components that could be damaged by the commonly available grippers require a specific grasping strategy and additional specific end effectors, thus increasing the complexity of the task.

These criteria are crucial to the feasibility of assigning tasks to robots and are part of the decision-making process in the offline allocation of tasks.

4.5.2 Flexible task allocation

Once the decomposition of the process based on the HTA guidelines has been completed, the next step is to perform a thorough analysis to coordinate collaboration at a higher level of hierarchy. Instead of a static task allocation, which would force the worker to adapt to a predetermined behavior of the robot, or a dynamic allocation, which may not fully synergize with human behavior due to unclear algoritmic decision-making procedure or lack of environmental information, an alternative is represented in mixing the advantages of both methods. Flexible task allocation enables dynamic and transparent task execution, placing the human at the center of the decision-making process. During the development phase, shareable tasks are designed so that they can be performed by both agents, and the cobot has to be programmed to flexibly adapt to the human's decisions and be able to execute the required actions. The criteria by which the human will decide which tasks to perform alone and which one to perform in collaboration with the robot are defined on the basis of cycle time, order size, cognitive and physical workload, and evaluated by the workers' knowledge and experience. In particular, the human operator should have a range of known options in an adaptive task allocation scheme to allow him to switch between different collaborative configurations as needed. Referring to the considered assembly task depicted in Figure 4.28, the subtasks can be considered as composed of two different categories. There are tasks that involve picking up aluminum bars and positioning them relative to each other, as well as fixing the angular joints to create the desired final structure. The human can perform both subtasks, but the three-dimensional shape of the structure forces the operator to hold the bars with one hand and fix the bolts with the other. In such a case, it would be helpful to delegate one of the two operations. To adhere to flexible task allocation and allow the worker to decide whether to request robot aid or not, the flow diagram of Figure 4.29 is developed. The chart can be considered divided into two main parts. The upper part refers to the first part of the task related to mounting and fixing the second bar with the first joint, and the remaining bottom part refers to the third bar and the second joint. At the beginning of the assembly task, the operator can trigger "Input 1" to require the robot to pick the first bar and bring it to the station remaining available for a collaborative hang-guiding manipulation, as illustrated in the left side of the diagram. The right part of the diagram refers to the case of a human assembly without the cobot. The second part of the task follows the same principle of on-demand robot assistance, considering whether the first part was performed collaboratively or not. In Figure 4.29 the blue rectangles refer to the actions of the human operator ([H]). The orange blocks refer to the robot ([R]). Green blocks indicate collaborative co-manipulation involving both humans and robots ([H + R]). The robot intervention in executing specific pre-programmed actions is triggered by human inputs indicated as "Input 1", for the first bar, "Input 2" for the second bar, and "Input 3" for the homing procedure after releasing any object grasped by the robot. "Input 4" begins and concludes the co-manipulation mode. All inputs are considered to be provided by the worker on will, allowing him to exploit the robot as a third arm capable of performing precise or heavy subtasks. The contribution of the human workforce can increase for small production quantities and decrease for larger batches.

To evaluate such an allocation algorithm, it is necessary to proceed with subjective and objective measurements. The objective measurements are related to performance values such as the completion time of the task and quality related values such as the correctly mounted final assembly or geometric consistency. Cycle time is the amount of time it takes to complete the task and can be considered as a measure of efficiency. On the other hand, defects in the assembly can lead to the final product not meeting its specifications, and these can be easily identified by inspecting the finished product.

We gather feedback from operators on manual and collaborative configurations using self-assessment tools. These include subjective questionnaires on perceived workload and emotional state after completion of the task.

Self-Assessment Tool for Operators

The NASA Task Load Index (NASA-TLX) is a popular, subjective, and multidimensional evaluation tool that measures perceived workload to evaluate the effectiveness of a task, system, team or other performance aspects. To assess the overall physical and cognitive workload, the NASA-TLX takes into account six subjective categories:

- Mental demand: represents the extent of cognitive and perceptual demand required to complete the task.
- Physical demand: describes the amount of physical effort needed to perform the task.
- Temporal demand: reflects the perceived time pressure due to the pace at which activities or task elements occur.
- Performance: refers to how well and how satisfied one is with the achieved results.
- Effort: describes how much a subject had to exert themselves (both mentally and physically) to achieve their level of performance.
- Frustration: reflects the degree of discomfort, stress, and annoyance experienced during the execution of the task.

Each category is expressed on a scale from 0 to 100.

Emotional state

The Self-Assessment Manikin (SAM) is a visual evaluation instrument used to measure an individual's emotional reaction to a particular situation or occurrence. SAM is used to collect the emotional state evaluation by assessing three dimensions:

- Pleasure: determines whether a feeling is pleasant or unpleasant.
- Arousal: describes an individual's level of excitement, regardless of whether it is caused by a pleasant or unpleasant feeling.
- Dominance: refers to the feeling of having control over a particular situation.

4.5.3 Experimental results

The experimental phase was conducted using the collaborative robot TM5–700 mounted on a work bench. The force/torque sensor, which was introduced in section 3.3.1, was installed between the robot and the gripper to allow precise gravity compensation and online hand guiding capabilities. This sensor detects the forces exerted by the operator and sends them to the cobot control system. This allows the cobot to respond in real time, adjusting its movement based on the forces applied. If the operator applies excessive or unexpected force, the cobot can stop its movement, thus helping to avoid collisions or damage. The force/torque sensor provides real-time feedback that enables the robot to modify its behavior accordingly. A finger gripper presented in 3.2.1 is used as an end effector to pick up and handle objects in the collaborative workspace. A button present on the built-in camera attached to the flange of the robot is used to convey command messages interpreted as inputs described in the flow chart of Figure 4.29. The inputs (ranging from 1 to 3) were determined by counting the number of times the operator pushed the button within a certain time frame. ”Input 4” is identified by a press and hold of 3 seconds.

The efficacy of the proposed solution is evaluated by having six people participate in the testing phase. To measure the variation in perceived workload, six university students aged 20 to 30 years, with similar body types and no prior knowledge of the task, were asked to assemble the structure. The four versions of the task included two manual assemblies without robot assistance, one with the standard configuration and the other with an extra mass of $1kg$ at the end of the last bar (as seen in Figure 4.28-4). The other two versions of the assembly operation allowed operators to use the cobot in HRC fashion, enabling them to assign the subtask as they wished. In this case, the assembly is also performed with and without the extra mass of $1kg$ at the edge of the third bar. Each variation of the task is executed six times by each of the six operators.

After each completed variant of the task, each operator completed the NASA-TLX and SAM questionnaires. The total perceived workload of the task is calculated by averaging the ratings of the six values expressed by the participants. Figure 4.30 illustrates the results obtained by analyzing individual responses. From the graph, it is possible to note how manual assembly operations with extra weight are recognized as significantly more mentally demanding. As expected, the physical effort required is also perceived to be higher in the weight-added configuration. A rise in time pressure,

as well as performance perception and effort, comes at the cost of a higher perceived frustration in completing the task with the extra weight in the case of complete manual assembly.
Taking into account collaborative execution (HRC), as is evident in the graph in Figure 4.30, the additional weight burden on the operator is almost entirely nullified across all dimensions. This was achieved through the collaborative robot's support in holding the bars, effectively neutralizing the disparities between the configurations. Comparing the four configurations (Manual, Manual + extra weight, HRC, HRC + extra weight), it is immediately evident that the manually executed configurations proved to be more demanding than those executed with the collaborative robot.

When comparing two manual configurations, one with extra weight and one without, it is expected that the overall sentiment would be less pleasant and more rushed. Additionally, the heavier bar and the difficulty of managing it can lead to a perception of lack of control. When examining the two configurations that were tested in HRC, it is evident that there are no significant differences between them in all aspects. When comparing the results between manual and HRC settings, the only significant differences found are related to general positive sentiment, finding not significant all other dimensions.

During the experimental phase, the cycle time of the tasks was collected for each execution as a reference to the time required to complete the assembly task. Regarding the mean values of the cycle time, no significant differences were observed ($t_{manual} = 162.9s$, $t_{manualw} = 178.8s$ in manual configurations and $t_{HRC} = 172.1s$, $t_{HRCw} = 164.0s$ in HRC configurations). Manual configurations have a slightly broader range of time distributions than those with the robot. This could be due to the robot's consistent speed, which leads to a more uniform assembly speed.

In manual configurations, 64.58% of the tests showed a deviation from the specified bar position, with an average displacement of $2.36mm$. In HRC settings, 36.67% of the trials experienced a discrepancy, with an average divergence of 1.67 millimeters. The operator's capacity to lock and unlock the robot to hold the components is responsible for the decrease in errors. This allows the operator to keep the bar at the designated height during the assembly process. Consequently, the errors that occurred were the result of the imprecision of the operator.

In conclusion, to validate the proposal for flexible human-managed task allocation, an HRC assembly task was tested by assigning it to individuals in the laboratory for data collection and analysis. The results showed a significant improvement in the perceived workload in the HRC configuration with respect to the completely manual operations. This was achieved through the complete elimination of perceived difficulties due to added weight. There were no significant differences in terms of average time, but a notable reduction in variability was observed, primarily due to decreased human variability with the use of the robot. For assembly errors, no significant differences were evident. Further investigation should explore other user interfaces to improve flexibility and efficiency for human-centricity in task allocation and awareness.

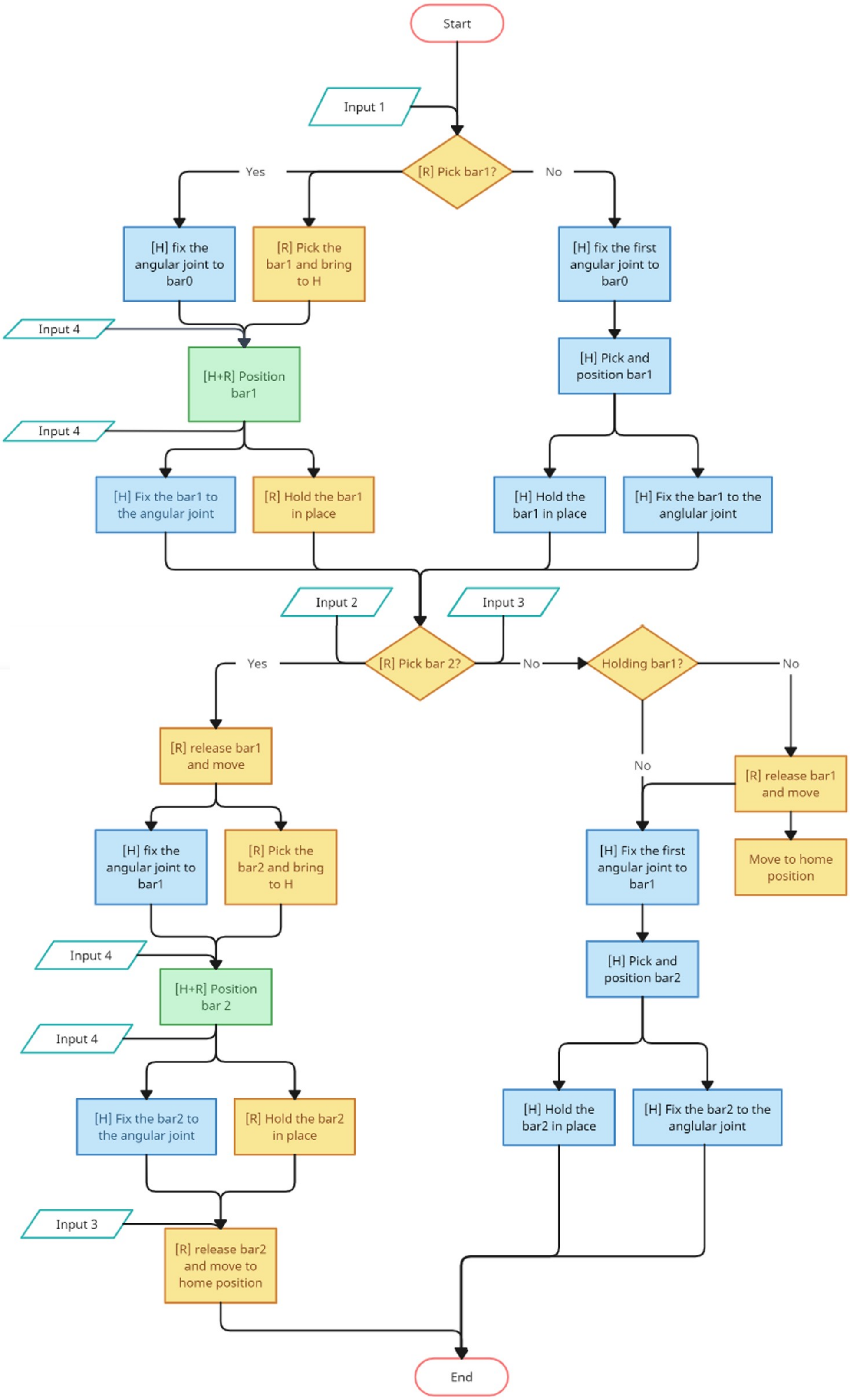


Figure 4.29: Flowchart of the flexibly allocated assembly sub-tasks. The blue rectangles refer to the actions of the human operator (H). The orange blocks refer to the robot (R). The green blocks indicate collaborative co-manipulation involving both human and robot(H+R).

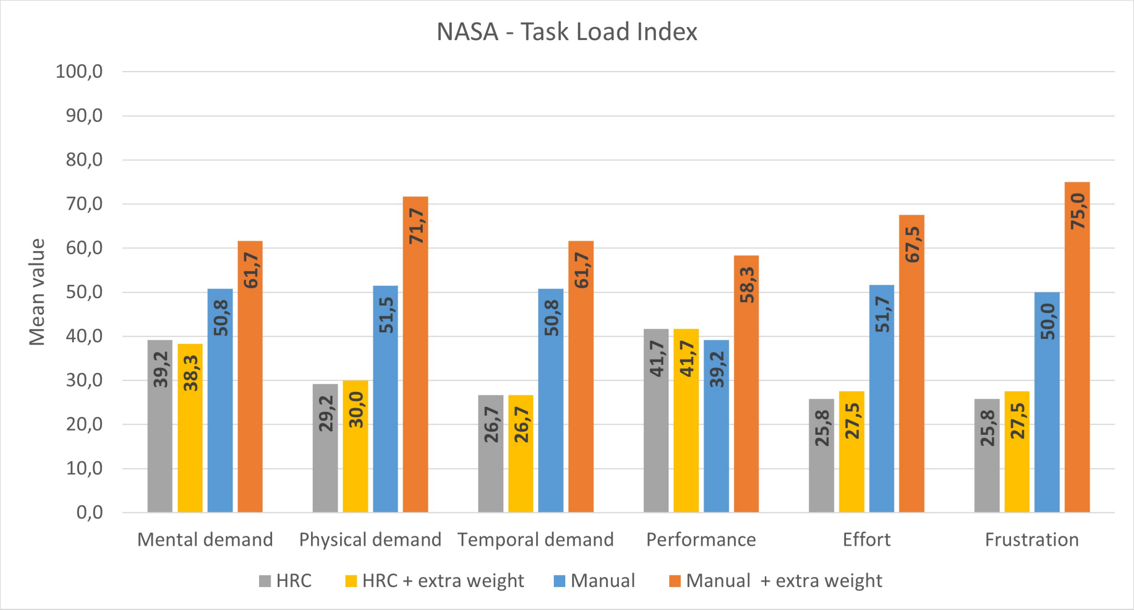


Figure 4.30: NASA - Task Load Index response of the six interviewed operators

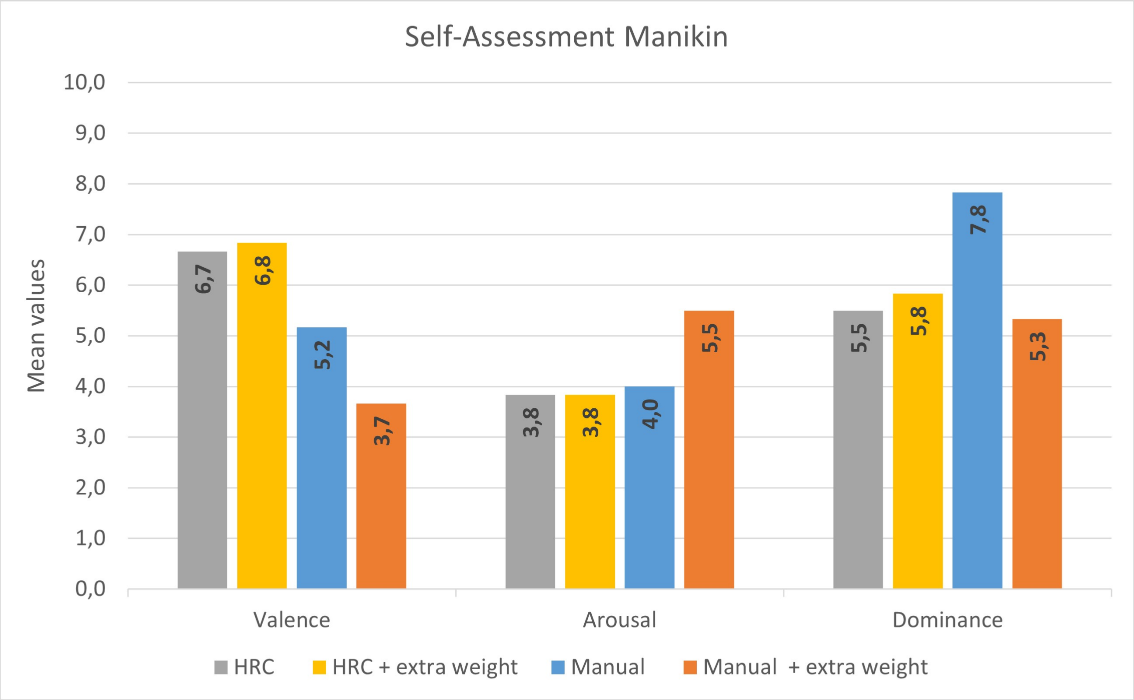


Figure 4.31: Self-Assessment Manikin response of the six interviewed operators

Chapter 5

Industrial applications

Collaborative robots provide a pathway to new applications that industrial robots are not able to access. They are ideal for craft-based, non-industrial settings or educational institutions, and are accessible to those who are not familiar with robot programming, such as artists, designers, and students. The simplified end-user programming of the cobots allows for a more flexible deployment of these machines. Originally, they were developed to contain the effort, complexity, and cost of robotic solutions, allowing the adoption of robots especially by small and medium enterprises. This chapter outlines industrial use cases that involve the utilization of collaborative robots.

The following section 5.1 focuses on the use of the most recent techniques for cobot programming and on how to manage their pros and cons when using them in an industrial setting. Simplified end-user programming, enabled by collaborative robots, aims to solve the issue of frequent intervention of highly specialized robot programmers, which can negatively affect the promising flexibility of these robotic solutions. Collaborative robots have been designed to reduce the effort and complexity of robotic solutions, allowing to implement flexible and lean automation that can be easily relocated and reprogrammed, thus reducing costs compared to a traditional robotic solution. These manipulators are ideal for the following purposes:

- Improve the efficiency and automate low value-added manual processes;
- Simplify work and reduce injuries;
- Work safely in a reduced footprint cell;
- Readjust easily the program to fit process variations.

To achieve these advantages, it is mandatory to understand the strengths and limitations of the collaborative capabilities of robots. It is important to note that cobots are not meant to replace industrial robotics that have been successfully used to increase the productivity and quality of production in many industrial settings. Instead, they represent an alternative solution in a specific target market, with different end users and different sets of needs. These necessities include higher flexibility and adaptation of the cobots to the rapidly changing processes. In such scenarios, avoiding the need for intervention by highly specialized programmers would improve flexibility and reduce the overall cost of the robotic cell. This brings to the need for simple and effective programming methods.

5.1 Cobot programming in industrial tasks

In [131], a review of intuitive robot programming environments for educational purposes was drawn, confirming the recent trend of using easy-to-program robots, mainly in the academic environment. In particular, the study focused on the benefits of visual programming languages (VPLs) and tangible programming compared to text-based programming languages (TPLs). However, according to the authors of [55, 132, 133], most of the research produced evidence that points to the success of their approaches with end users, although only a few of them conducted a formal experiment. Moreover, the test subjects were mostly students with little to no programming skills in a university setting, as opposed to typical users in a real setting. Considering this, it is evident that VPLs need to be evaluated with real end-users as opposed to university students, and such evaluation should be validated in a real setting as opposed to a laboratory. It is vital to examine how end-user developers extend, debug, and deploy applications created with visual programming tools throughout the application life cycle.

The aim of the present chapter is to report a direct experience of a visual programming language used to program a collaborative robot for a real-case industrial application. The programming procedures, together with the design process of the robotic cell are described in order to provide a real setting evaluation of the simplified robot programming method.

In the following, the developed application is depicted, focusing on the relevant details of the industrial end-of-line deployment of a cobot. The goal of the tasks and the technology used provides a complete overview of the application and the main reason for using a collaborative solution. The layout of the robotic cell is revealed and the details of the automation process are described. The programming key points are then discussed with respect to the features proposed by the literature. The structure and characteristics of the final program are analyzed to detect the challenges and advantages of using a collaborative programming approach.

5.1.1 Industrial case description

The robotic case study proposed in the present chapter is an end-of-line unload of capsules to seal wine, sparkling wine and champagne bottles produced by an automated industrial machinery. These machines are modular, versatile, and have simple and low maintenance management, allowing one to quickly reset the production process. The capsules, made of aluminum or polylaminate, can be printed in hot foil, flat, or embossed. They can be made with complex decorations such as embossing, punching, tear-off tabs, etc. The capsules produced are collected in a single row of 60 to 80 items, depending on their dimensions and the setup, as shown in Fig.5.1a. Rows of capsules represent the final product of the production line. Each machine is provided with a conveyor belt with ridges, which serves as buffer, capable of collecting 8 final capsule rows before letting them fall into a collection container. The unloading of the final products is carried out manually by an operator who takes the rows of capsules, one by one, and places them on a tray. This is the operation that needs to be robotized in the described application, to develop an automated end-of-line. Plastic-made trays are available in two different sizes of 6, as in Fig.5.1b, or 7 slots for the rows of capsules. Subsequently, they are stacked on top of each other to build a pallet of 30 to 40 trays. The trays on the pallet are alternated in size and shifted relative to each other, as in Fig.5.1c, to reduce the height of the filled pallet and, at the same time, to avoid damage to the product. The palletizing steps are depicted in Fig. 5.1.

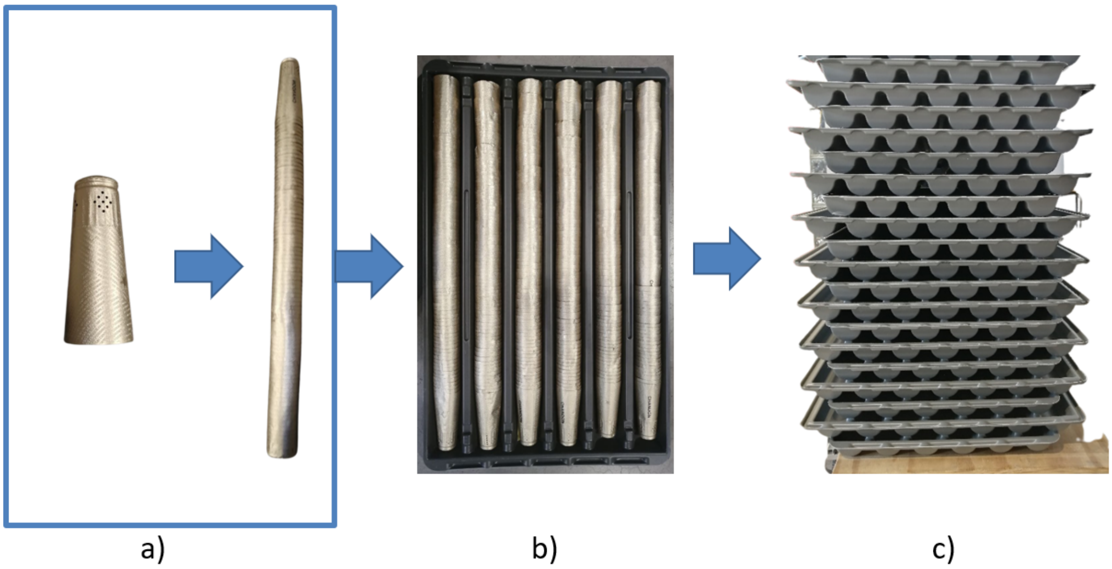


Figure 5.1: Steps of palletizing process. a) The final products of the automated machine currently are the rows of capsules; b) the capsules are placed into plastic trays of suitable shape; c) The trays are stacked on the pallet in alternating sizes of 6 and 7 slots for capsules rows.

The machine cycle time is long enough to not continually require the operator's intervention, allowing him to perform alternative tasks while the buffer is being filled. The characteristics of this manually performed task make it a perfect candidate for an automation process. In fact, the task is characterized by tray movements and displacement of the rows of capsules, being therefore a non-ergonomic, physical, low-value-added process which forces the operator to work in fits and starts. The operator performs a palletizing task which consists of a series of pick-and-place subtasks that can be easily accomplished by a robot that has been programmed correctly. Moreover, considering that the product may vary frequently over time and that a reduced footprint is a desirable feature for a company, it is clear that a robotic flexible solution fits the job perfectly. Figure 5.2 represents the manual setup of the unloading cell. In the foreground is visible the pallet of empty trays on the left and on the right the corresponding output stack of trays to be filled with the products falling in the buffer collector of the machine. In the background, the operator in charge of unloading the machines can be seen performing the task on the second machine.

Figure 5.2: Manual setup of unloading cells. In the foreground is visible the pallet of empty trays on the left, and on the right the corresponding output stack of trays to be filled with the products deposited in the central buffer collector of the machine. In the background, the operator unloads the second machine with similar characteristics.

Cobots are machines that are user-friendly and versatile, they can sense their environment and align with other machines without much effort or recalibration. They can be easily equipped with additional components, such as extra axes, grippers, force sensors, and communication devices, through a plug-and-play system. After analyzing the production task, the environment, and possible robotic solutions, the chosen robot model is an OMRON TM12 equipped with two appropriately designed grippers, which can be exchanged on demand at the change tool station.
Due to flowchart-based programming, an intuitive human-machine interface (HMI), and simple, hand-guided teaching functions, almost no prior programming knowledge is required to start programming this machine. The user can build complete function blocks and fill in the predefined attributes with the desired parameter values. This robot is the ideal choice for the situation due to its safety features and the potential to utilize its integrated vision system. This collaborative solution is the only feasible option to allow workers to enter the robotic cell for quality control or to optimize execution by adding their field experience to the application.

5.1.2 Developed robotic task

After analyzing the manually performed unloading process, using RoboDK simulation software, it was possible to explore the various layout configurations of the final robotic cell. The final layout, which allows the robot to perform the task within the cycle time limit of capsule row production, is shown in Fig. 5.3. The robotic manipulator is placed on a supportive structure from which all relevant points are reachable. There are 2 pallets in the robot proximity: the robot has to pick a tray

from the feeding position of empty trays (label:1 in Fig. 5.3) and move it to the second pallet (label:2 in Fig. 5.3). Regarding the second pallet, the robot must correctly position the trays to alternate them, as required in manual operation and described in Fig. 5.1 c.
Due to this need, also the feeding column of the trays (label:1 in Fig. 5.3) has to be correctly set, alternating the trays of 6 and 7 slots. This raises a challenge in correctly identifying and selecting the desired item from the feed location due to possible mismatching or errors. Therefore, the position labeled 3 in Fig. 5.3 is foreseen to serve as a stack of discarded trays in case of need.

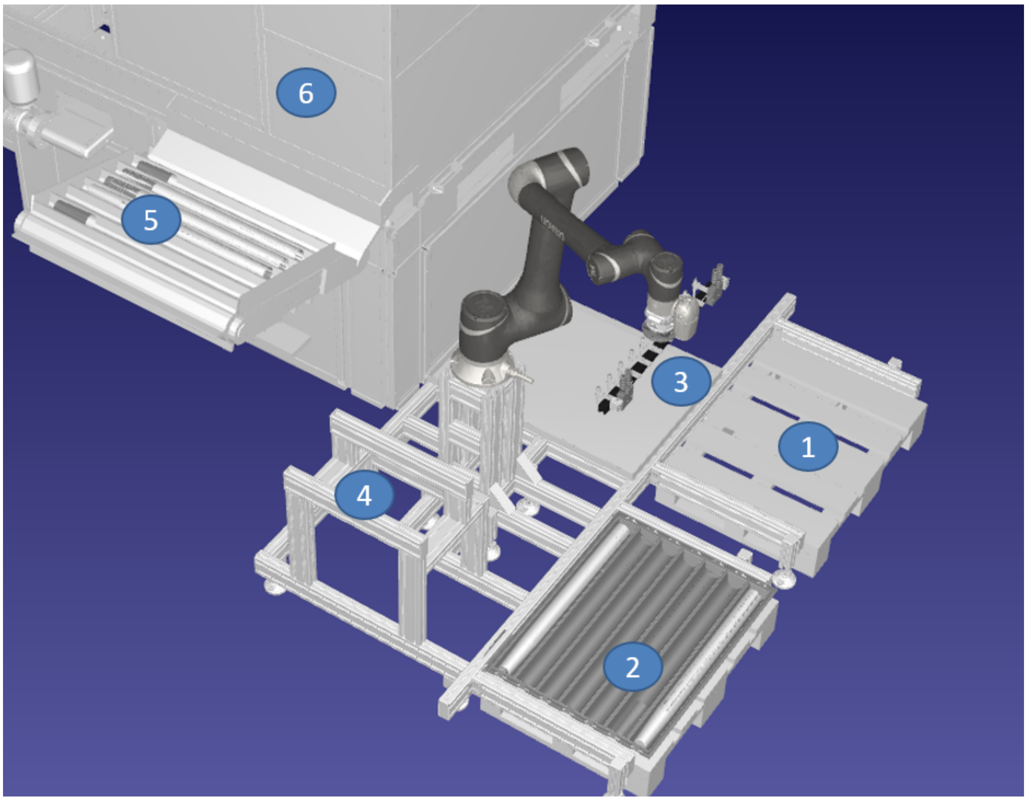


Figure 5.3: Setup of the robotic application. Labels are, respectively: 1) trays feeding pallet; 2)output pallet of palletizing application; 3) discarded trays position; 4) tool change station; 5) conveyor belt buffer for the capsules row; 6) capsules rows production machine.

The tray identification process is carried out using the integrated camera on the OMRON TM12 that provides the information about the type and relative position of the tray to pick. This information is obtained from the barcode applied on each tray. Since the depth information is not detectable using only the 2D camera, it is extracted from the sonic sensor, mounted on the dedicated gripper for trays, through an analog signal. The interior of the tray is not flat, and the sonic sensor is only able to give a rough estimation of the distance between the tray and the suction cups of the gripper. Therefore, after a coarse initial movement, the last few millimeters are traveled in a compliant, force-sensing movement that is enabled by the cobot. In fact, the TM software provides a function which is capable of compliantly sensing a given force limit when the robot moves along a single direction. This setting is used

for searching for objectsects and recording the coordinate values at the triggering touch-stop position. This approach eliminates the issue of the stacked trays not being completely rigid, which could cause the vacuum cups to not properly attach to the inner curved surface of the trays' slots. For the sake of robustness, the grippers are equipped with an electronic vacuum switch for grip checking. Once the tray is properly picked up from the feeding station, the placement operation is trivial, as the relative position of the palletized trays is known.
Before being implemented on the real setup, the reachability of all the pick and place points has been evaluated in the RoboDK simulation. All movement and joint configurations were tested to correctly select the position and orientation of the robot base. Figs. 5.4a and 5.4b show respectively the lowest and highest possible positions of the stacked trays in the feeding position, Fig.5.4c and Fig.5.4d the highest and lowest in the tray placing positions.

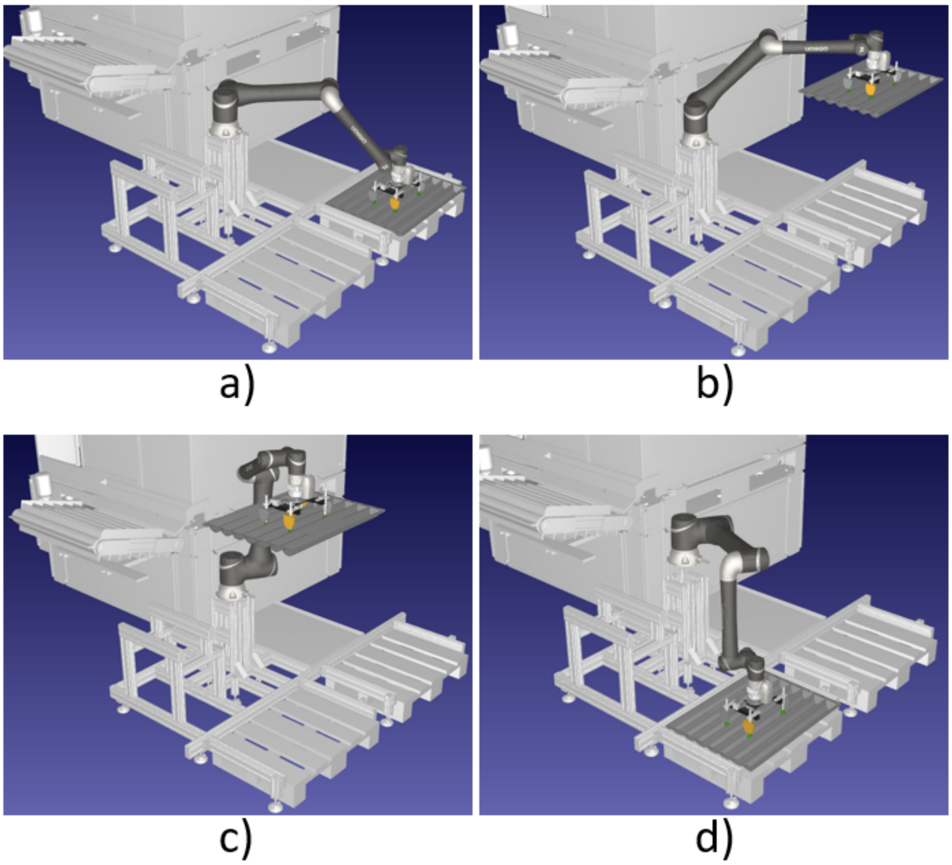


Figure 5.4: Trays pick and place procedure simulation. In Figures a) and b) the robot picks the tray from the feeding pallet trays at different heights. Figures c) and d) represent the first and last tray placing positions

Once the empty tray is correctly positioned, the cobot can start filling it with the rows of capsules. Since the shape of the capsule row is completely different from the trays, the gripper depicted in Fig.3.10a, used for the first part of the task, is no longer suitable. Another vacuum gripper with different numbers, configurations, and sizes of mounted cups is available at the tool change station labeled 4 in Fig.5.3. The

capsules gripper, represented in Fig.3.10b, is designed to meet the features of the capsules row surface.
While the cobot executes the first part of the palletizing routine and the tool exchange, the machine (label 6 in Fig. 5.3) is allowed to carry out the production process, accumulating the rows on the conveyor belt (label 5 in Fig.5.3).

The buffer is reachable by the cobot at every position, but is made available only if the conveyor is not moving, to avoid any collisions of the gripper with the ridges of the belt. To achieve such synchronization, an exchange of consents between the robot and the PLC driving the conveyor is required. The latter manages communication with the machine upstream of the production cycle. Once consent is received, the cobot picks up the capsule row from the conveyor belt (Fig.5.5a), moves in the position over the available tray (Fig.5.5b) and places the row in an empty slot (Fig.5.5c). The cycle closes with the robot retraction (Fig.5.5d) to continue the loop until the tray is full. If the last slot has been occupied, a new tray has to be placed on top of the last one filled. If the last tray has been filled, the robot requires human intervention for output pallet substitution. To this end, subdivining the robotic workspace into two main areas allows the worker to enter the shared environment for quality control, maintenance, or any other intervention without triggering emergency alarms. The area related to capsule production, labels 5 and 6, is safeguarded with laser barriers able to prevent the robot entering that portion of the cell in front of the machine output window, offering human access for any reason. On the other hand, the remaining area is accessible by requesting the consent of the robot by pressing the entering request button. If such a request is detected, the robot, after completing the ongoing portion of the task, assumes a safe neutral pose, leaving the zone unobstructed for human access. Any unforeseen access to the cell will immediately stop the robot movement.

At any time, the cobot keeps track of the number of rows unloaded, discarded trays, and correctly displaced ones. It has to be aware of when and if the operator's intervention is necessary, and warn about the incoming or triggered halt of the system. To meet these requirements the cobot has to be programmed accordingly, considering the necessary communication protocols, error handling routines, and sensors readings.

The advanced features introduced in the robotic cell are not trivial to program even if the flowchart-based interface of the collaborative robot is used, especially if the program has to be designed from scratch. To illustrate a practical industrial application of collaborative robot deployment and assess the key characteristics of the simplified robot programming technique, the structure of the flowchart-based program is outlined below. Relevant characteristics such as nesting, parallelization, and looping are contextualized to highlight those elements that improve the robustness of the application, the expressiveness in the creation of a flexible program, the generalizability to different scenarios, and the overall readability.

5.1.3 Robot program structure

In this section, the developed application is discussed in the light of the literature review regarding simplified robot programming, to point out the most relevant

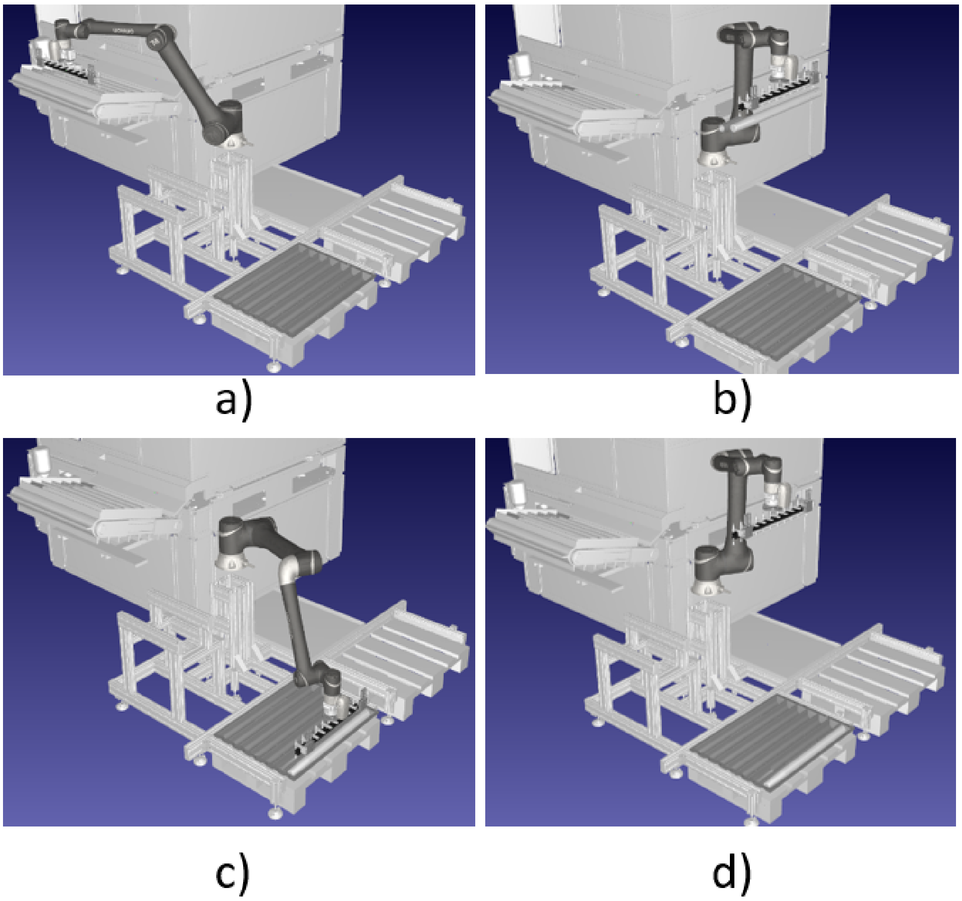


Figure 5.5: Capsule rows pick and place procedure simulation. a) The robot picks up the row of capsules from the conveyor belt. b) The robot approaches the placing position. c) The capsule row is placed in the empty slot of the available tray. d) The robot retracts to start the next loop of the capsule pick-up and place procedure.

elements that a programmer of any level should consider during the design process of the program. All the programming phases should be considered during the design process, including authoring, editing, and debugging tasks, as well as the physical nature of robot programming. As stated in [134], the authoring phase of any robot program should consider the following aspects:

- authoring scope: it is mandatory to understand all the specific requirements of the process to be automated, fulfilling all the necessities;
- identification of the robot capabilities: the advantages and the limitations of given robotic solution have to be considered and possibly compared to some alternatives;
- programming method: some of the robotic solutions may offer various ways of programming such as flow chart, text or cad based methods. Others foresee a wizard guide or allow for the lead-through and walk-through programming. The choice of the programming method is relevant also for the editing phase and in all the cases when it is foreseen a future program modification;

- programming features: are all those features which allow the program to be the general, flexible and understandable in order to allow easy future changes. The features that would help to improve the readability and simplify the overall structure of the robot program are:
  - nested and recursive programming
  - lean loops management
  - parallelization of subprograms
  - event driven structure of the conditional instructions
  - detailed high-level error handling

As stated in[135], the expert manual operator may know the process much better than an expert robot programmer. The authors report that before starting the programming phase, it is necessary to analyze the overall production process and subsequently split it into lower-level tasks. This operation is often performed with the help of the process expert, which has to be consulted various times before and after the program creation. The program outline should follow this abstraction, nesting the subroutines into a convenient scheme.
The structure of the application developed in this work can be subdivided into two main activities of picking and placing items and the two necessary gripper change routines. A ”waiting” subprogram is foreseen in the case of the required operator's intervention. One last routine is built to initialize the robot when powering up and perform important initial checks. The same initialization routine addresses the recovery procedure of the last step executed by the robot. The overall scheme is depicted in Fig. 5.6.

The OMRON TM12 used in the proposed application is programmable via TMFlow, a graphical HMI that provides end users with a simple interface for robot motion and logic programming environments. Through the HMI, users can manage and set the robot parameters to plan movements and process logic. Despite the intuitiveness of the flowchart-based programming interface, the design of a robust industrial application from scratch is far from being trivial. Starting from the initialization process, it is necessary to continue adding conditional checks to correctly start or restart the robot task execution. As an example, the represented initialization procedure in Fig.5.7 may be subdivided into 3 main parts:

1. A global variables check is executed at the start up. It is necessary to adapt the behavior of the robot with respect to all the conditions previously met:
   - the robot is close to the conveyor belt;
   - output pallet is full or close to be so;
   - how many trays are filled on the output pallet;
   - what size is the last placed tray;
   - how many slots are empty on the last placed tray;
2. The analyzed variables will trigger a different recovery behaviour which has to be evaluated in a getaway node;

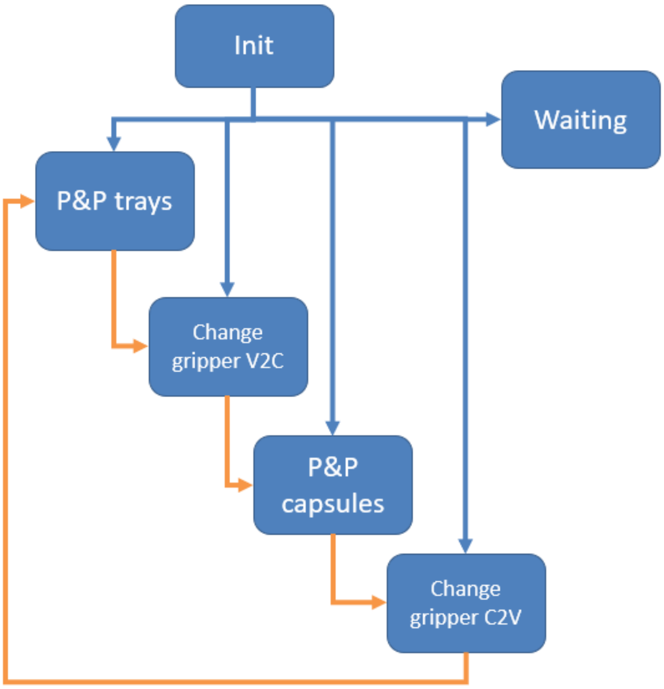


Figure 5.6: Structure of the programmed task

3. The initialization ends by recalling a sub-task keeping track of the initialized variables.

To understand the effectiveness of TMFlow VPL, it is necessary to point out all the relevant characteristics that a programming environment should present to effectively help the end user write a simple, robust, and safe program. According to [136], the missing milestone in introducing simplicity in robot programming is not only a matter of simplifying complex programming languages or low-level technical issues, but offering an easy programming environment that allows users to instruct a robot from a high-level programming interface. In [136], the authors propose an interactive programming approach based on natural spoken language and block-based interaction to be applied in a fixed sequence. Such an approach resulted to be highly structured and mainly driven by the system, lacking capillary support to the user in designing correct and safe programs.

The capabilities of identifying missing objects, actions, or locations and suggesting how to complete the program are difficult to achieve without losing the generality of the programming environment.

In the case of TMFlow visual programming, the creation of the program logic is completely supported by the flowchart-like representation of the program, in which each functional block is connected to one or more other blocks. TMflow allows one to follow the program execution steps, highlighting the block in progress. However, if the flow chart is too tangled, it may be very challenging to debug or find the parameter to modify for optimization or correction purposes. A post-authoring reorganization of the scheme is required to improve the expressiveness of the flowchart. Loops, nested parts and subprograms, decision nodes, and parallel routines should be easily identifiable to allow the editing and debug phases. Moreover, the physical task steps

of the robot program should be evident in the final scheme to allow skill identification and conditional branching.
The developed subtask of tray displacement, depicted in Fig.5.4, is characterized by a large number of logical elements to correctly pick and place the items and perform the necessary error checks. The built-in vision system, the sonic depth sensor, and the vacuum switches are logically implemented in the tray pick procedure while being clearly identifiable. The corresponding TMflow scheme is depicted in Fig. 5.8, where the single branches correspond to different parts of the logical procedure of the subtask. TMFlow software does not allow the inclusion of notes on the chart; therefore, it is essential to have a well-structured arrangement of the branches. Following the enumeration of the branches, it is possible to identify the function of each group:

0. Recovery node;
1. Initialization and variables check;
2. Approaching movement to the pick position using the sonic depth sensor;
3. The initialization guide the sequence in evaluating if at this point the next pallet is 6 or 7 slots pallet, and so if the first part of the barcode visual recognition process is necessary or not;
4. Barcode recognition is performed to get the information about the type and the relative position with respect to the reference pick point;
5. The last approach sequence uses the touch stop function of the TM12 to sense the contact of the gripper with the tray surface. Small adjustments are performed once the vacuum pumps have been activated and until the vacuum switch returns a digital signal meaning that the tray is correctly attached;
6. During the displacement, the vacuum switch continuously check the presence of the attached tray until it is correctly placed on the output pallet;

4e. If the barcode does not correspond to the expected one and so the alternation of the two types is not respected, a check is performed to ensure that the following tray is to discard;

5e. If the previous condition is verified same procedure as in 5) is executed;

6e. The discard position is reached and the tray is released;

7e. The robot returns to the initial pick point and restarts from the 2* branch.

During the execution of the trays pick and place tasks a parallel thread continuously exchange information about the status of the robot and the machine. Communication with the PLC that drives the conveyor belt is established through the MODBUS protocol.

To improve the robustness of the program, a recovery handling step is implemented to recover the last state of the task and correctly initialize the variables after a reboot of the system. It is the step zero of every sub-program, as is reported also in Fig.5.8, in brach 0. This step is executed for all subtasks in order to rely on a recovery routine independently from the step at which the robot may encounter an error or a shutdown. This practice also allows one to easily identify any undesired behavior in the robotic application for further debug action or process modification.

5.1.4 Results discussion

An analysis of an industrial end of line manual palletizing process was conducted and a robotic automation solution was created. This allowed the operator to be relieved of a physically demanding low-value task that was causing him to work in short bursts. The dull manual process has been replaced by a sequence of pick-and-place subtasks that can be completed accurately by a correctly programmed robot. The developed application was first designed and simulated using RoboDK software and then implemented using a collaborative OMRON TM12 robot. The resulting robotic cell is depicted in Fig.5.9. The implemented barriers have various access points safeguarded by laser barriers in order to allow a fluid supply of empty trays, easy collection of the final stack of filled trays, access to the automatic machine from all sides, and access to the capsule buffer for quality control.

The programming phase was carried out through use of the TMflow software which is a flowchart-based robot programming environment. The features of the developed task were described and the challenges highlighted considering the recent research argument of simplified robot programming.

Considering an industrial environment, for initial development of a complex task the VPL is not the best in terms of simplicity and requires a deep understanding of the process to automate.In contrast, the upkeep and troubleshooting processes, alterations and editing operations, as well as the general comprehensibility of the designed task are significantly enhanced when compared to text-based programming techniques. The first attempt to program a full industrial job is still demanding, even when utilizing simplified programming techniques. Despite the fact that a collaborative robot is used, a novice-level user may not be able to develop a robust application without help. However, very little effort is required to understand an existing program and be able to modify and maintain it.

Considering the task developed, the use of a collaborative robot opens up a wide range of possible improvements to be implemented. Increased security of the application paves the way for many optimizations for process quality and cell layout, unlocking the possibility of close human-robot interaction. Under nominal conditions, the human task shifts towards the three main operations depicted in Figure 5.10. The operator is required to supply the pallets of empty trays to the robotic cell, supervise the starting phase of the machine, thus checking the quality compliance to the requisites of the products being produced by the automatic machine, and prepare the output stack of trays for shipment. The workers are relieved from the dull and repetitive job of displacing the capsules in trays, reducing the physical effort related to the task. Mental demand is kept low due to the advanced safety sensors and algorithms implemented in the robotic station. The operator can gain access to multiple sections of the workspace while remaining sure of compliant and safe robot behavior, in accordance with the safety standards.

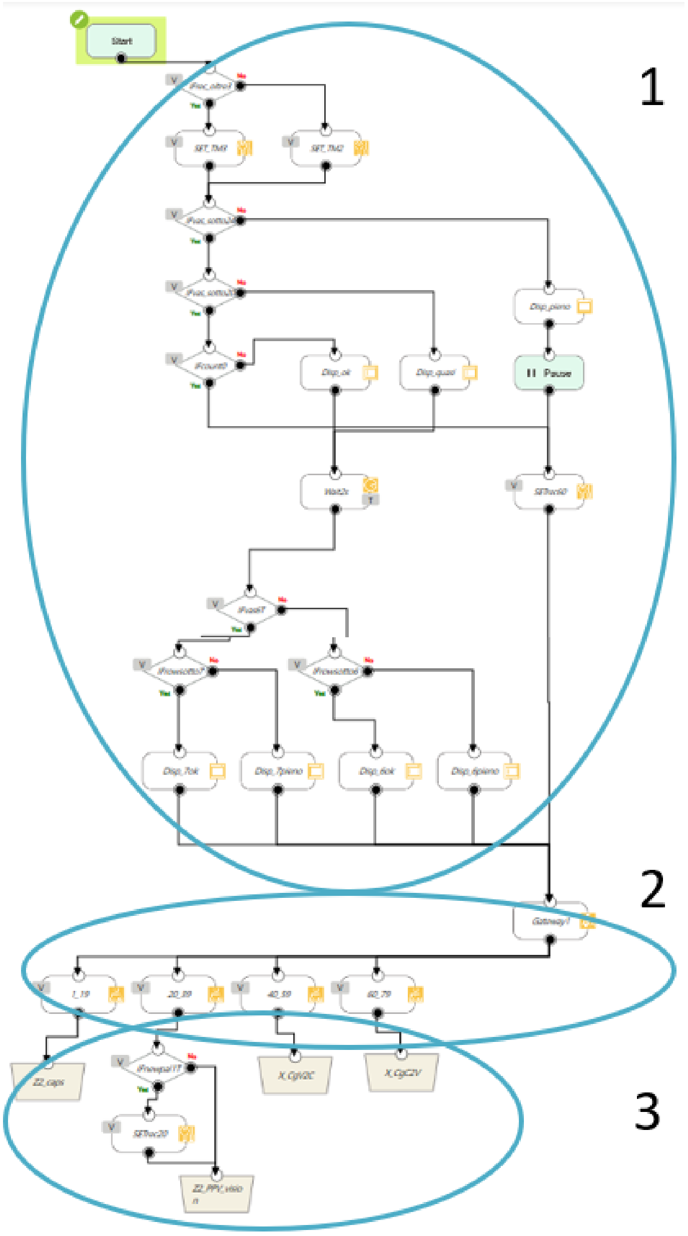


Figure 5.7: Initialization subprogram scheme

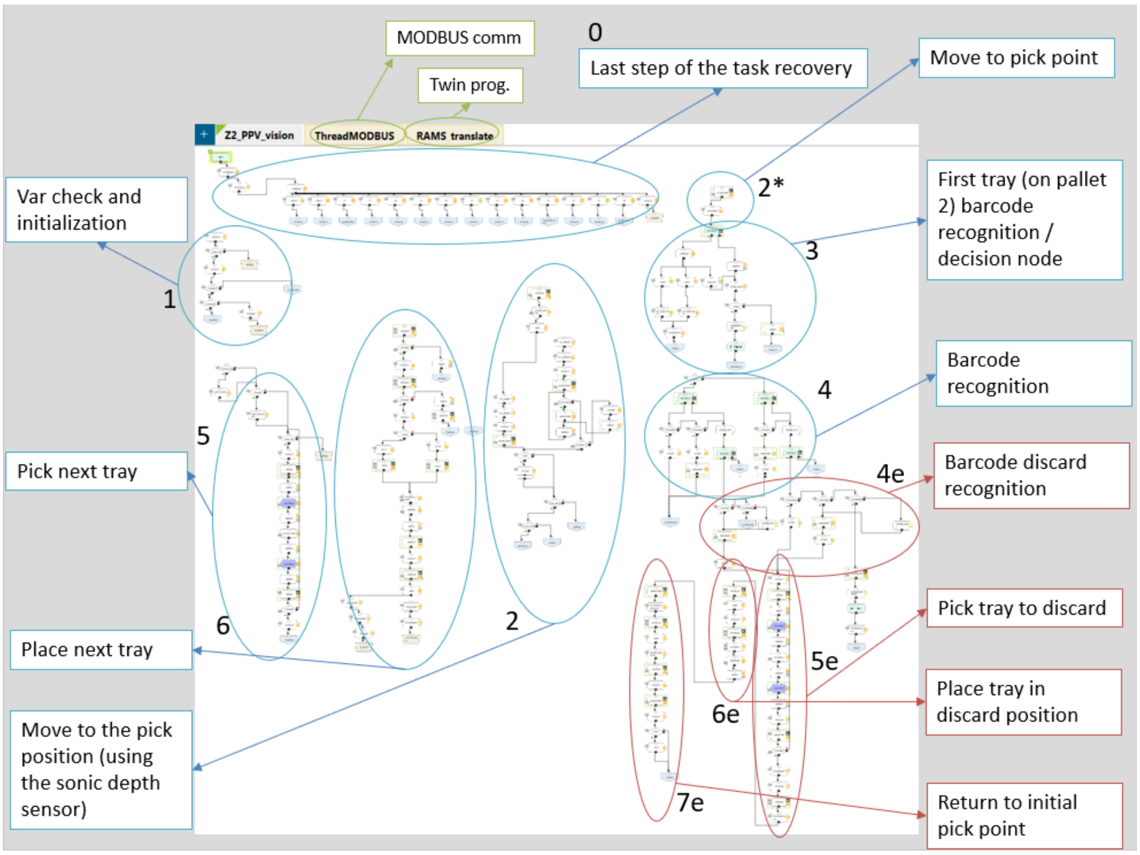


Figure 5.8: Trays pick and place task scheme in TMFlow visual programming HMI

Figure 5.9: The developed robotic cell.

Figure 5.10: Operator's duties after implementing the robotic unloading cell. The operator is required to supply the pallets of empty trays to the robotic cell (left), supervise the starting phase of the machine, thus checking the quality compliance to the requisites of the products being produced by the automatic machine (center), and prepare the output stack of trays for shipment (right).

5.2 Enhancing flexibility and safety while empowering workers: alternative layout

The manufacturing landscape is rapidly evolving and is today characterized by the emergence of Industry 5.0, leading to a growing recognition of the importance of a human-centered, sustainable, and resilient approach. In response to these needs, collaborative robotics is considered a promising solution to enhance flexibility in industrial operations, while ensuring a higher level of safety. Cobots enable human-robot collaboration with reduced physical barriers, thereby reducing the footprint of the robotic station and fostering a closer working relationship between humans and machines.

This section focuses on the alternative layout proposition for the application of collaborative robotics in end-of-line industrial operation, specifically in the unloading of products from an automatic machine introduced in the previous section 5.1. By managing repetitive tasks such as placing the trays near the end of the line of the automatic machine, waiting for them to be filled, and placing them on a trolley which will then be moved by the human operator, the robot significantly improves efficiency. This automation frees workers from low value-added operations and allows them to supervise and manage multiple machines simultaneously, thus enhancing productivity. The approach behind this proposal is to keep robotic tasks simple, easily reprogrammable, and modular, while empowering the human operator to participate in supervision, decision-making, and problem-solving tasks. This collaboration allows the operator to assume a supervisory role, organizing production based on identified needs, and promoting their participation in production line management tasks.

The effectiveness of this approach lies in leveraging the advantages associated with promoting the capabilities of human operators and creating a meaningful relationship between humans and robots. By harnessing the power of collaborative robotics in end-of-line industrial operations, thus allowing increased flexibility and shorter response times to market, manufacturing processes can become more agile, adaptable to changing market requirements, and therefore improve competitiveness.

5.2.1 Contextualization

Industry 4.0 brought about a significant transformation in the manufacturing sector, taking advantage of advances in automation, data exchange, and digital technologies. It revolutionized traditional factory operations by introducing intelligent systems and machines, paving the way for increased efficiency, productivity, and flexibility. As technology continues to progress together with society and economy, the Industry 5.0 paradigm has emerged to further redefine the manufacturing landscape. End-of-line manufacturing processes are a crucial domain where Industry 5.0 can demonstrate its transformative potential, particularly in optimizing the packaging and shipment of products and directly influencing supply chains. Traditionally, the automation of these processes has relied on fixed robotic cells that incorporate safety measures such as fences and protective lasers to safeguard human workers. This approach limits the agility of end-of-line processes, which often require reconfiguration and additional safety measures when adapting to changing product requirements or

introducing new products. This is why, when flexibility is the main requirement, end-of-line operations are nowadays still carried out manually by human operators. In contrast, a fenceless layout empowered by collaborative robots would present a more versatile and adaptable solution, potentially reducing the cost effort to modify the manufacturing setup and saving precious footprint areas.
Some of the key end-of-line processes that can be automated using robots include:

- Palletizing: the process of arranging products onto pallets for efficient storage and transportation;
- Packaging: task of sealing, labeling, and sorting products into appropriate containers or boxes;
- Quality control: perform quality inspections, checking for defects, inconsistencies, or deviations in products, in order to enable early detection and prevents faulty products from entering the market;
- Sorting and distribution: sort and distribute products based on predefined criteria, such as size, weight, or destination, to facilitates the smooth flow of products within the supply chain, reducing errors and enhancing order fulfillment;
- Material handling: loading and unloading items onto conveyors, transferring products between workstations, or replenishing supplies.

The adoption of collaborative robots in end-of-line manufacturing processes presents opportunities for increased flexibility, seamless human-robot interaction (HRI), and reduced operational costs, since the operator is free from repetitive tasks and can manage and supervise multiple machines simultaneously. These advancements contribute to the realization of Industry 5.0 principles, creating a harmonious and efficient collaboration between human workers and intelligent machines.

The aim of this section is to present an application of the above-mentioned approach, leveraging human-centric aspects such as operator empowerment and human supervision, with reference to the design and deployment of a fenceless robotic cell at the end of line of an automatic machine for the production of sealing capsules for bottles. The objective is to provide valuable information on the potential benefits and considerations associated with implementing the Industry 5.0 principles in the manufacturing industry, while emphasizing the integration of human workers and collaborative robots. In the following subsection 5.2.2, relevant related works are discussed and general guidelines for collaborative robotic design are drawn. In section 5.2.4 the working principles of the proposed layout are reported. Finally, conclusions and considerations are discussed in section 5.2.5.

5.2.2 Motivation

Extensive research has been conducted to examine the cost-oriented, reconfigurable, and flexible features of production systems in relation to cobot capabilities. However, the authors of [12, 137] highlighted a significant knowledge gap in training methods for collaboration and the use of cobots in the manufacturing industry. Their analysis pointed out that, contrary to expectations, the end-users described most cobot applications as limited to low-level interactions, such as pressing start/stop buttons, with minimal flexibility in deployment. Furthermore, experts expressed the need for traditional robotic skills to interact and collaborate effectively with cobots. This analysis revealed the importance of the concentrated efforts of HRI researchers to investigate and assess enhanced design approaches in this domain [138]. The possibility of substituting the manual material handling process at the end of the line of an automatic machine that produces capsules for bottles has been previously introduced in section 5.1. In the proposed solution, the material handling process has been completely replaced by an automated sequence of pick-and-place subtasks, performed by a suitably programmed cobot. The cobot has to unload the rows of sealing capsules collected on a buffer conveyor at the end of the automated machine and place them in a tray. When a tray is filled up, the capsules pick and place task is interspersed by the positioning of an empty tray on top of each filled one, in order to pack a pallet of 40 filled trays. When the pallet is ready, the robot waits for the operator to remove the finished package using a pallet truck and provide a new empty pallet in the correct position. Once everything is reset, the worker communicates to the robot through the HMI the permission to restart the unloading and palletizing processes. The proposed layout considered two different end effectors and an exchange station to allow retooling. The final solution foresees a robotic cell that requires some fences and laser barriers to identify an unannounced entry of the operator and to trigger the protective stop of the robotic movement. At the same time, the possibility for operators to safely enter the robotic cell is guaranteed, since they can communicate their intention to enter the cell through an HMI (screen and button). In such a case, the robot moves to a safety position, allowing the operator free access to the automated machine, so as to carry out any needed check. The operator can enter the cell with the robot operation on standby, and the task can be restarted safely after his/her exit.

As a further step towards the Industry 5.0 paradigm, the present section describes a new layout for the above-mentioned application, designed to enhance the human involvement in the automated process. A different task allocation is also presented, in order to allow the worker to undertake his supervisory role.

5.2.3 Industrial task description and layout update proposition

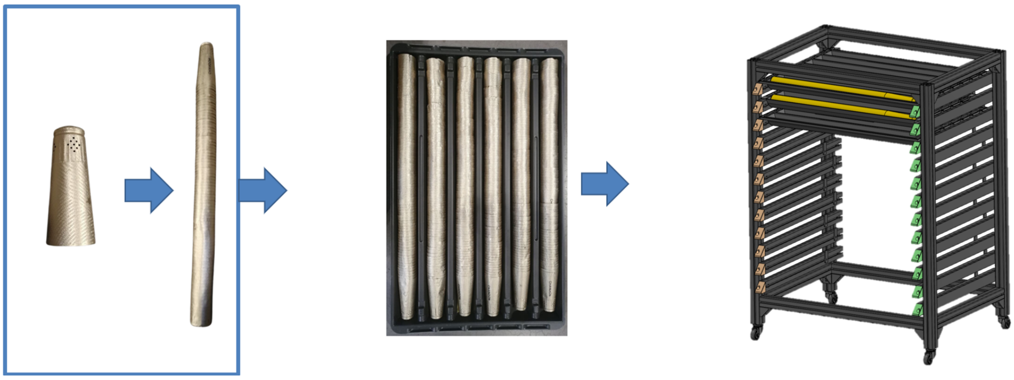
As already mentioned, the proposed robotic case study in this research pertains to the unloading process at the end of the production line. The manufacturing process is identified in the production of sealing capsules for bottles using automated machinery. Production line machines are efficiently managed with low maintenance requirements, allowing for quick processing resets. To maintain this advantage of the production line, obviously, automated unloading processes also have to be flexible. This means provide to the process the resilience characteristics foreseen in industry 5.0 paradigm while exploiting the worker's expertise, decision-making and critical capabilities, allowing him a higher level of control on the timing of the production.

The capsules considered are manufactured from aluminum or polylaminate materials and can be customized with various decorative features, including hot-foil printing, flat or embossed designs, and additional elements such as embossing or punching. The capsules are collected in a single row, as shown in Fig. 5.11a, typically comprising 60 to 80 items depending on their dimensions and the specific setup. These rows of capsules represent the output of the production line before the end-of-line robotic cell. To facilitate the workflow, each automatic machine is equipped with a conveyor belt featuring ridges that serves as a buffer, in order to decouple the unloading and palletizing operations from the production. This buffer can hold up to eight rows of capsules before releasing them into a collection container.

The unloading task of the final products was originally performed manually by an operator who carefully placed each capsule row on a tray, as in Fig. 5.11b. Since no modifications in shape are allowed, to preserve the form of the malleable capsules, the latter are stored in plastic trays. The trays used in this process come in two different sizes. There are trays with six slots for the rows of capsules, and another with seven slots. The filled containers are placed on a pallet in an alternating pattern of six and seven slots to conserve space and protect the product from harm.

Automating the palletization operation was the primary objective of a previous section, discussed in [139], with the aim of developing an automated end-of-line solution. In this section, we explore a different design that takes into account the participation of employees in the cycle management and monitoring activities in line with the human-centric approach proposed by the new Industry 5.0 paradigm. As previously described in 5.1.4, the robot requests pallet substitution when the stack of trays is ready, forcing the human intervention at a specific moment in the process, stopping the unloading operations while. The aim of the new layout is to minimize downtimes by shifting the pace management of the process from the robot to the operator. In this section, the proposed solution is to provide a couple of rack trolleys in which the trays are picked up and placed in order to minimize the interruptions in the robot operations. This configuration enables seamless transitions for the robot between trolleys, eliminating the need for unnecessary stops and ensuring continuous operation, while the worker can easily transport away the wheeled trolleys (see in Fig. 5.11c), having some flexibility in deciding when to perform the operation. This solution frees the operator from the times dictated by the machine, thus allowing him to oversee and manage multiple machines simultaneously.

A couple of grippers for trolleys are mounted on the support structure connected to the robot, to ensure the trolleys correct positioning and avoid movements during the tray insertion operation. The top view of the overall robotic cell is depicted in Fig. 5.12. It was decided to supply the automated robotic station with trolleys containing empty trays at the beginning of the unloading process. The robot picks each tray, presents it to the conveyor in the correct position to receive the row of capsules into each pit, exchanging consents of movement with the buffer, and finally returns the tray to its original location in the trolley. The responsibility of managing the exchange of trolleys and replacing the filled with the empty ones is assigned to the worker in charge of a supervisory task. However, the trolley exchange can happen at any moment after the robot filled it and passed to the next. By decoupling the unloading process from the production line and the trolley exchange, the operator



(a) (b) (c)

Figure 5.11: Steps of end-of-line unloading process: a) The final products of the automated machine are the rows of capsules; b) The capsules are placed into plastic trays of suitable shape; c) The trays are placed in a rack trolley for an easy transportation.

is allowed to decide its scheduling autonomously and to provide assistance to the robotic cell whenever he is free. Moreover, this solution foresees only one gripper and allows to avoid the gripper exchange station, since only the trays are moved. By simplifying the process, we reduce the possible source of breakdowns.

To be able to pick the trays from the trolley, the robot has to be equipped with a specially designed end effector. Tray supports have been foreseen to ensure proper insertion into their allocated positions in the trolley, as they must maintain their straight shape and avoid any bending. Additionally, a proximity sensor is required on the gripper to check the actual presence of the tray during task execution and initialization. The designed robot end effector is represented in Fig. 5.13. The only moving part is the one that grips the tray and is moved by two pneumatic cylinders.

The robot used for this application is an Omron TM12. This manipulator represents a collaborative robotic solution tailored to a wide range of automation tasks, including the unloading procedure under investigation. It encompasses advanced functionalities and attributes that make it a good fit for secure and efficient HRC. Characterized by its compact and lightweight design, TM12 integrates seamlessly, has a maximum load capacity of 12 kg and up to 1300 mm reach, offering the required flexibility to access various positions within the workspace. The TM12 integrated vision system enables object identification, shape recognition, and Optical Character Recognition (OCR). Collaborative capabilities ensure that interactions with humans in shared environments are conducted safely, with controlled force and speed. The robot programming is facilitated by an intuitive and user-friendly software interface, empowering operators to quickly configure and customize tasks. The TM12 cobot incorporates the essential capabilities and functionalities necessary for the unloading application under investigation. The build-in vision system is used to recognize a marker of a different shape for each tray size in order verify that the specific tray is present in a given location and that tray is of the required size.

It is crucial to ensure that the position of the robot base in relation to the buffer and trolleys aligns with the dimensions of the trays being moved, as well as the robot's own movement requirements. Once all the necessary elements are prepared

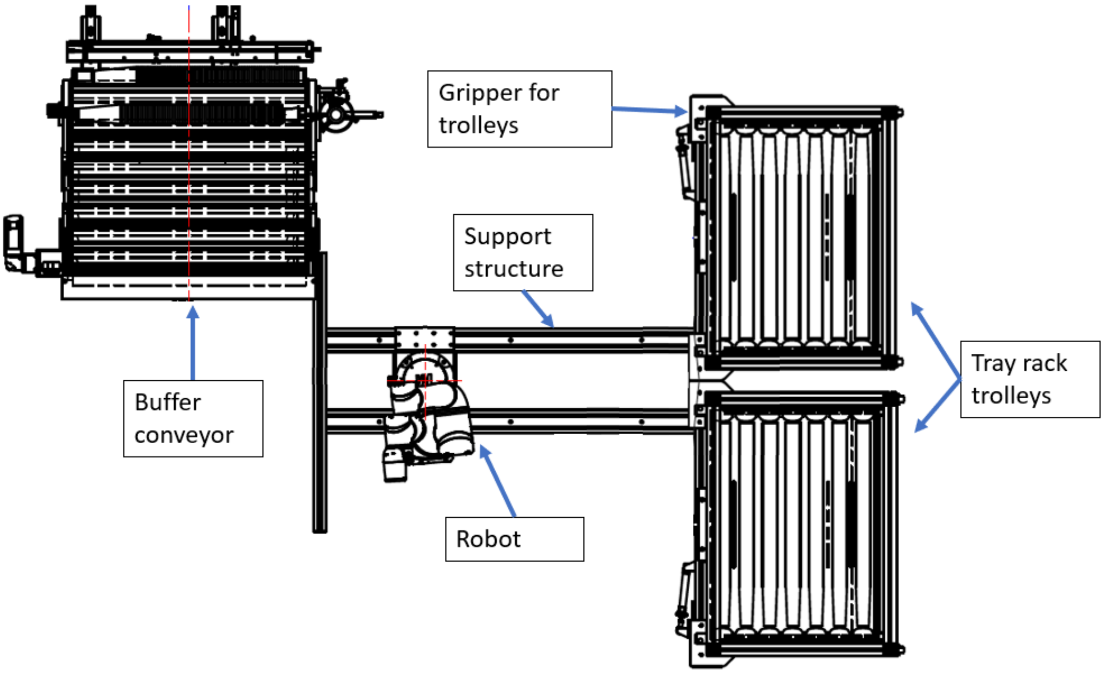


Figure 5.12: Top view of the robotic station layout

for layout evaluation, a simulation of the task is performed in an environment that allows us to assess potential collisions and identify the optimal trajectory for the robot to follow, as in Fig. 5.14a.

After verifying the layout in the simulation environment, laboratory tests and evaluations are performed, as depicted in Fig. 5.14b, to validate the proper functioning of the system. This process includes implementing and executing debug procedures to identify and resolve potential issues or errors. Additionally, thorough testing of the sensors is conducted to ensure their accuracy and reliability in detecting and responding to relevant environmental signals. These testing and evaluation procedures are crucial for verifying the system's performance, according to the production cycle time, and ensuring its smooth operation before proceeding to the next stages of implementation and deployment.

To guarantee the reliable execution of the robotic task, a proximity sensor is installed on the robot end effector. This sensor plays a crucial role in verifying the presence of the tray at the grip and release points, as well as at key checkpoints throughout the task. By continuously monitoring whether the tray is securely held by the gripper, the proximity sensor enables the system to react appropriately to unexpected failures or deviations, ensuring a robust and accurate execution of the task. To ensure trolley detection and facilitate seamless operation of the system, two additional proximity sensors are strategically placed on the support structure, one per each trolley, as can be seen in Fig. 5.15a. These sensors verify the presence of trolleys within the dedicated position. When the trolley gripper opens after complete handling of an entire trolley, as in Fig. 5.15b, the same sensors are interrogated to confirm its removal. In addition to the operator's confirmation, the signal from the sensors is expected to exhibit a distinct pattern of falling and rising. This pattern

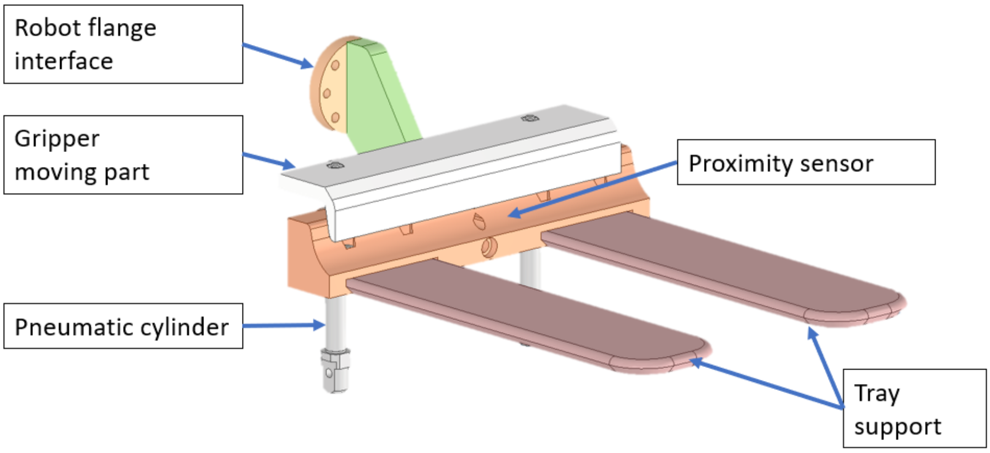


Figure 5.13: Gripper designed to carry the trays

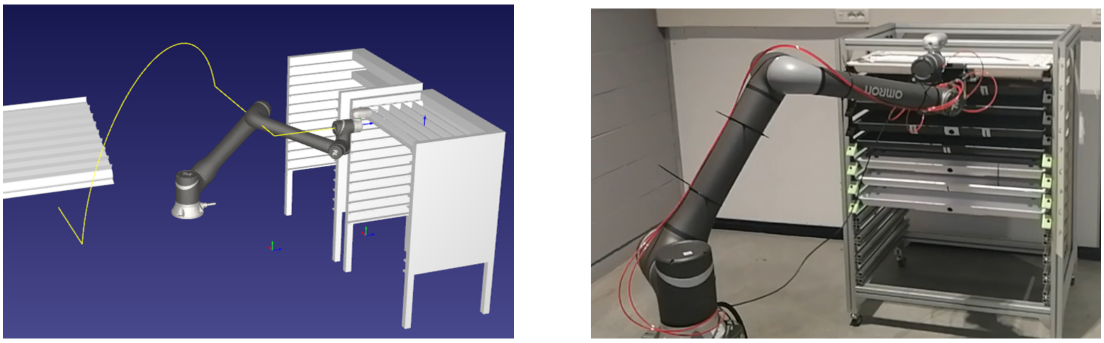
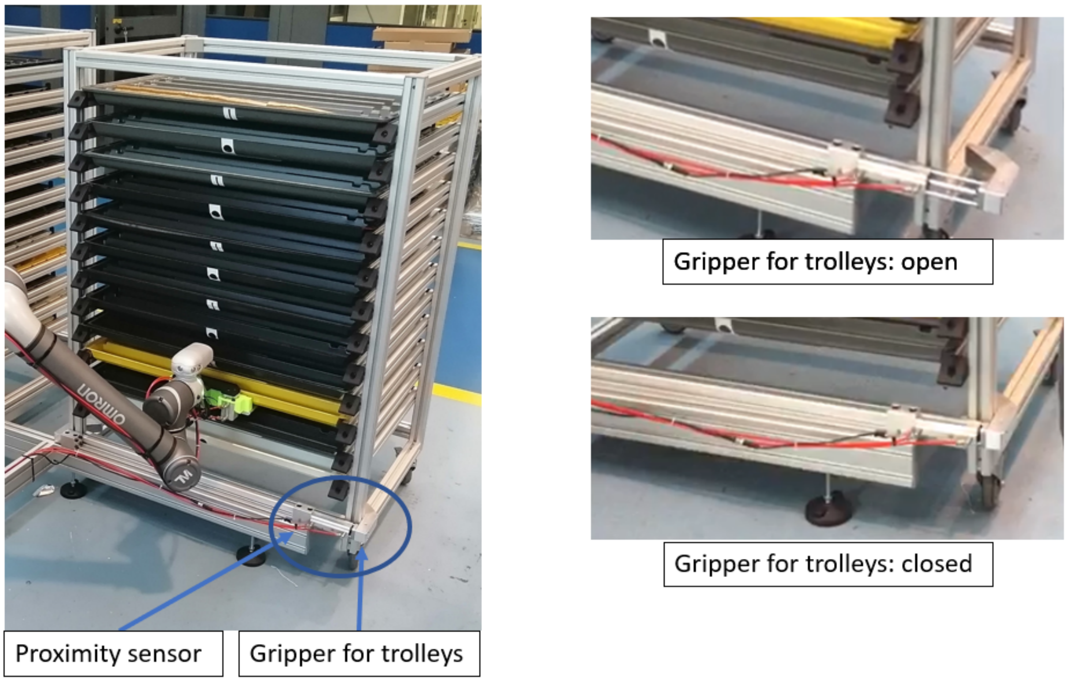

(a) *(b)*

Figure 5.14: Layout testing and validation a) Simulation environment for collision check and trajectory optimization; b) Laboratory test for robot programming, sensors integration and validation.

indicates that one trolley has been removed while another one has been replaced. This check serves as a flag for the robot controller, signaling it to close and securely hold the trolley in place while resetting the associated variables related to that specific trolley position. By incorporating this functionality, the system ensures accurate tracking of trolley movements and facilitates the smooth and efficient operation of the overall process.

5.2.4 Robotic task execution

The robot task starts with the robot in a rest almost vertical position, as depicted in Fig. 5.16a, away from the buffer conveyor or the trolleys, waiting for the worker to request the execution of the unloading task. When automated unloading is required, the robot passes to the activation state and communicates with the buffer conveyor to perform a check on various components such as the line and robot states variables, the presence check of the trolley and tray, and the input of the operator. If no errors occur, the robot starts the automated unloading cycle.
The manipulator moves in front of the first available tray of the trolley to be filled,

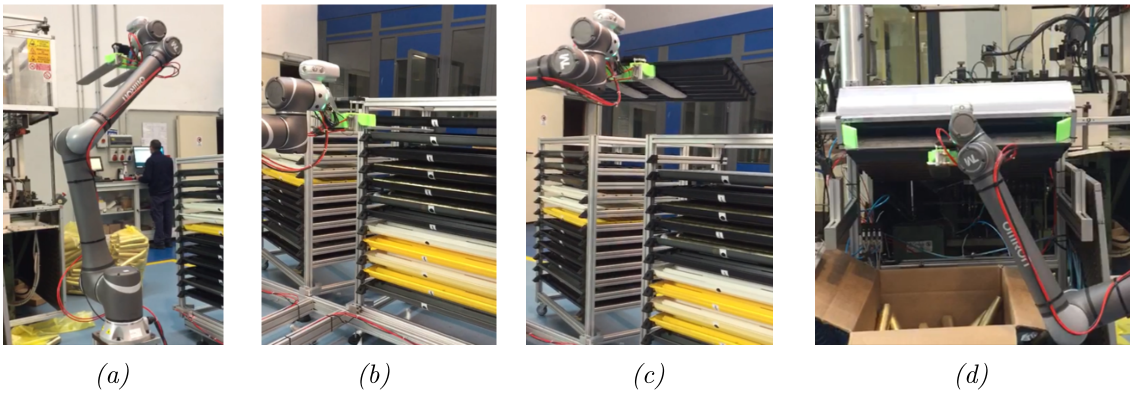


(a) *(b)*

Figure 5.15: Gripper and proximity sensors to verify the presence of the trays rack trolley. a) Overview of the correctly placed trolley in the dedicated position of the workspace. b) Detailed view of the trolley gripper functioning

as in the example in Fig. 5.16b, and checks whether the tray is present and its size. The verification process is performed by analyzing the acquired image of the label placed on the tray, using the built-in vision system. Once the presence of the tray is ensured, the end effector is moved in the pick position to grip the tray and extract it from the trolley, activating the pneumatic cylinder of the moving part of the tray gripper. From this moment on, and until the robot re-positions the tray, the proximity sensor will be queried by the end effector to ensure the presence of the tray during movement. The design of the end effector will hold the tray in a straight position as in 5.16c, preventing it from bending. The tray will be presented in front of the conveyor, as represented in Fig. 5.16d, and after the consent of the PLC that manages the conveyor, the tray will be moved in order to match the first empty row of the tray with the ejection position of the buffer. After the consent exchange, the buffer will move in order to eject a row of capsules, which will roll in the presented empty slot of the tray. The conveyor PLC will trigger the robot to move the tray in one row, to prepare the container to receive another row of capsules.
Once filled, the tray will be brought back to the same position where it was taken and the task will iterate for all trays of the trolley. During the execution of the task, the robot maintains continuous communication with the production line about the current tray and the capsule row. If the last tray of the trolley is filled correctly, a trolley change is required. In case of availability of a second trolley, the robot will continue filling the trays of the empty one, unlocking the filled one for substitution.

The robotic task is managed by a series of states and actions to ensure efficient



(a) (b) (c) (d)

Figure 5.16: Task description

operation during the unloading process. The states in which the robot can be are briefly described in the following:

1. Neutral state: the robot is positioned away from the trolleys and the buffer conveyor, in a ”neutral position” and waits for an activation signal from the line. In this state, the manual operation is allowed and the production continues considering manual unload of the capsules rows.

2. Activation state: the system prepares for operation by checking the status of various components, including the buffer, robot conditions, trolleys’ presence, variables related to the last picked tray, and operator’s inputs. Based on these checks, the next state is determined: either ”In Function” or ”Error.”

3. In function state: the robot executes the unloading cycle, maintaining continuous communication with the line to indicate the current tray and capsule rows number, while continuously exchanging consents for the advancement of the conveyor. If ”last tray condition” is triggered while picking the last tray, the robot communicates to the line, and hence to the operator that the current tray needs to be replaced to avoid interrupting the unloading cycle. The next state depends on the specific conditions: ”Trolley change required”, if a replacement is required after filling both of the available tray rack trolleys, ”Maintenance”, if raw material replacement is needed by the production line, or ”Error” if any unforeseen condition stops the execution of the unloading process.

4. Trolley change state: if both trolleys are filled, the robot moves to the ”neutral position” that allows for trolley replacement and awaits for confirmation signal from the operator. However, after filling the first, if a second one is available, the robot starts loading the second trolley, opening the trolley gripper and allowing the substitution.

5. Maintenance state: when the production line signals a downtime for any reason, the robot is required to reach the ”neutral position”, allowing maintenance operations in all the working space. If the robot is underneath the conveyor, while waiting for the next row of capsules to be delivered, it returns the tray to its place in the trolley and moves to a position that allows the operator to

move freely in all the work area. The controller waits for an activation signal from the operator to continue the interrupted process.

6. Error state: If an error occurs, the robot communicates the error code to the line and saves the error log and the current state of the system for recovery. The next state can be "Neutral", to allow for a manual unloading and keep the robot away from the buffer conveyor, "Activation", to restart the process, or "Error" if further investigation or action is required.

Collaborative robotic cell management allows the worker to operate the unloading process, switching from manual to automated material handling mode with respect to needs or its own decision. Improved flexibility promotes the operator to a decision maker and supervisory role while ensuring an agility gain due to smart pace management of the cell.

5.2.5 Final considerations

Collaborative robots offer numerous benefits, especially when contextualized in the Industry 5.0 environment. By leveraging such enabling technologies like cobots and enabling strategies such as HRC and HRI many advantages can be achieved in both efficiency of the production as well as the improvement of the working environment where human operators are present. One of the key advantages is observed within smart cell setups, where collaborative robots equipped with sensors and advanced vision systems enable safe and efficient navigation and interaction. This improves productivity and minimizes the risk of accidents or injuries.

By delegating physically demanding or repetitive tasks to cobots, human operators experience reduced physical strain, leading to improved well-being and long-term health. This allows workers to allocate their energy and focus on supervisory tasks that require human judgment and decision making, facilitating the promotion of operators to higher-level skills jobs. This empowerment allows workers to develop new skills, coordinate activities, and take on supervisory roles, thereby enhancing their expertise and contributing to their career advancement. The proposed collaborative workstation is designed to be easily editable, agile, and reconfigurable with respect to the production needs. This unleashes a virtuous spiral in which the on-demand usage of collaborative robots allows quick adaptation to changing tasks or production requirements.

The layout proposed in this work exploits the unlocked advantages of a collaborative robotic application that focuses on the human-centric approach suggested by the new industry 5.0 paradigm. In particular, in order to avoid downtime due to robot requests for pallet substitution, described in 5.1, and allow the human operator to manage the supply of empty trays, setting the timing of his operations, a double trolley position is implemented. We propose a solution that involves picking and placing the trays in a pair of rack trolleys alternately, with the aim of giving the worker more flexibility to transport the wheeled trolleys as needed, deciding when to perform the substitution of filled trolleys with empty ones. This solution liberates the operator from being bound by machine-defined timelines, empowering them to oversee and manage multiple machines simultaneously while minimizing interruptions in robot operations. In fact, this configuration enables smooth transitions for the robot between the trolleys, eliminating the requirement for unnecessary stops and enabling uninterrupted operation. By decoupling the robotic unloading process from the production line and the trolley exchange, the operator gains the autonomy to independently schedule their tasks and provide assistance to the robotic cell whenever they have availability. This decoupling empowers the operator to efficiently manage their workload and allocate their time effectively, enhancing the overall productivity of the system.

As a future improvement of the collaborative workstation, collision avoidance and human detection capabilities based on advanced vision systems would greatly increase the safety of the collaborative cell and allow task optimization. On the other hand, allowing the robot to automatically calibrate its position with respect to the trolleys or the buffer conveyor positions, would allow using mobile robot bases, increasing the overall agility and resilience of the robotic solution.

Chapter 6

Conclusions

This thesis has investigated the transformative journey of collaborative robotics from its early industrial applications to the emerging era of human-centered collaborative cells. The inception of industrial robots revolutionized manufacturing processes, enhancing precision and productivity while reducing risks for human workers. However, their integration came at a high cost and was limited to large-scale production facilities, leaving smaller companies and dynamic environments underserved.
The advent of Industry 4.0 and the evolving Industry 5.0 paradigm have shifted the focus from purely automated systems to collaborative processes that involve humans and robots working together. This shift is driven by the recognition that human expertise, agility, and adaptability are essential in industries where tasks are complex and rapidly changing. Cobots have emerged as a solution to bridge this gap, offering safer and more flexible interactions between humans and robots.

Throughout this thesis, various aspects of collaborative robotics have been explored. Safety considerations were discussed, highlighting the importance of continuous risk assessment and effective communication in environments where humans and cobots coexist. The need to address these complex challenges is pivotal to the success of collaborative robotics. Furthermore, the potential impact of automation on artisanal professions was explored, emphasizing the importance of recognizing the complementary nature of handcraft and automation. This synergy can foster innovation, diversity, and resilience in the craft sector, promoting sustainability and inclusion.

The development and adoption of cobots have introduced changes not only in technology but also in standards and regulations, with organizations like ISO updating norms to accommodate these transformative technologies. This thesis has investigated human-robot interaction techniques and provided guidelines for efficiently deploying cobots within the Industry 5.0 framework. The presented research covers practical applications of cobots in tasks such as glue deposition and assembly. These applications have demonstrated the feasibility of cobots in improving productivity and reducing variability in industrial processes. In addition, the integration of brain-computer interfaces into collaborative workstations was explored, showcasing the potential for hands-free control of robots through brainwave signals.

Although the thesis highlights the promising prospects of collaborative robotics, it acknowledges several challenges, including human operator dependencies in some

tasks. These challenges present areas for future research and development.
In summary, this thesis highlights the pivotal role of collaborative robotics in shaping the future of automation, emphasizing the importance of human-robot collaboration. Addressing safety concerns, considering artisanal professions, and harnessing the potential of advanced human-machine interfaces, pave the way for a more efficient, inclusive, and innovative industrial landscape in the Industry 5.0 era. Collaborative robots are not merely tools for automation, they are catalysts for positive change, empowering both workers and industries to thrive in an evolving technological landscape.

List of peer-reviewed publications

1. Castelli, Kevin, Marco Gavioli, Yevheniy Dmytriyev, e Hermes Giberti. ≪Mechanical Design and Development of a Continuous Rotational Variable Stiffness Actuator≫. In 2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), 1–6. Ankara, Turkey: IEEE, 2021.
   https://doi.org/10.1109/HORA52670.2021.9461398.

2. Castelli, Kevin, Ahmed Magdy Ahmed Zaki, Yevheniy Dmytriyev, Marco Carnevale, e Hermes Giberti. ≪A Feasibility Study of a Robotic Approach for the Gluing Process in the Footwear Industry≫. Robotics 10, fasc. 1 (31 dicembre 2020): 6.
   https://doi.org/10.3390/robotics10010006.

3. Dmytriyev, Yevheniy, Marco Carnevale, Hermes Giberti, e Giovanni Todeschini. ≪On Cobot Programming in Industrial Tasks: A Test Case≫. In 2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), 1–9. Ankara, Turkey: IEEE, 2022.
   https://doi.org/10.1109/HORA55278.2022.9800018.

4. Dmytriyev, Yevheniy, Federico Insero, Marco Carnevale, e Hermes Giberti. ≪Brain–Computer Interface and Hand-Guiding Control in a Human–Robot Collaborative Assembly Task≫. Machines 10, fasc. 8 (5 agosto 2022): 654.
   https://doi.org/10.3390/machines10080654.

5. Dmytriyev, Yevheniy, Ahmed Magdy Ahmed Zaki, Marco Carnevale, Federico Insero, e Hermes Giberti. ≪Brain Computer Interface for Human-Cobot Interaction in Industrial Applications≫. In 2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), 1–6. Ankara, Turkey: IEEE, 2021.
   https://doi.org/10.1109/HORA52670.2021.9461383.

6. Zaki, Ahmed Magdy Ahmed, Yevheniy Dmytriyev, Kevin Castelli, Marco Carnevale, e Hermes Giberti. ≪Trajectory Planning for Contact-Based Robotic Applications by Use of a 3D Stereo Depth Camera≫. In 2021 3rd International Conference on Robotics and Computer Vision (ICRCV), 68–72. Beijing, China: IEEE, 2021.
   https://doi.org/10.1109/ICRCV52986.2021.9546958.

List of submitted papers

7. Dmytriyev, Yevheniy, Marco Carnevale, and Hermes Giberti. ≪Enhancing flexibility and safety: collaborative robotics for material handling in end-of-line industrial operations≫. In 2023 5th International Conference on Industry 4.0 and Smart Manufacturing (ISM 2023).
   Abstract:
   The manufacturing landscape is rapidly evolving, being characterized today

by the emergence of Industry 5.0, which leads to a growing recognition of the importance of a human-centered, sustainable, and resilient approach. In response to these needs, collaborative robotics has emerged as a promising solution to enhance flexibility in industrial operations, while ensuring a higher level of safety. Cobots enable human-robot collaboration without the need for physical barriers, thereby reducing the footprint of the robotic station and fostering a closer working relationship between humans and machines.

This article focuses on the layout proposition for an application of collaborative robotics in an end-of-line industrial operation, specifically in the unloading of products from an automatic machine. By managing repetitive tasks such as placing the trays near the end of the line of the automatic machine, waiting for them to be filled, and placing them on a trolley which will then be moved by the human operator, the robot significantly improves efficiency. This automation frees the workers from the low value-added operations and allows them to oversee and manage multiple machines simultaneously, thus enhancing productivity. The approach behind this proposal is to keep robotic tasks simple, easily reprogrammable, and modular, while empowering the human operator to engage in supervision, decision-making, and problem-solving tasks. This collaboration allows the operator to assume a supervisory role, organizing production based on identified needs, and elevating their involvement to production line management tasks.

The effectiveness of this approach lies in leveraging the advantages associated with promoting the capabilities of human operators and creating a meaningful relationship between humans and robots. By harnessing the power of collaborative robotics in end-of-line industrial operations, thus allowing increased flexibility and shorter response times to market, manufacturing processes can become more agile, adaptable to changing market requirements, and therefore improve competitiveness.

Bibliography

[1] Chris Turner and John Oyekan. “Human-Centric and Sustainable Industry 5.0: Application to Holonic, Flexible, Reconfigurable and Smart Manufacturing Systems”. In: *Sustainability* 15 (June 27, 2023). DOI: `10.3390/su151310169`.

[2] Esben H Østergaard. “Welcome to industry 5.0”. In: *Retrieved Febr* 5 (2018), p. 2020.

[3] Doriana M. D’Addona, Fabrizio Bracco, Andrea Bettoni, Nariaki Nishino, Emanuele Carpanzano, and Alessandro A. Bruzzone. “Adaptive automation and human factors in manufacturing: An experimental assessment for a cognitive approach”. In: *CIRP Annals* 67.1 (2018), pp. 455–458. ISSN: 00078506. DOI: `10.1016/j.cirp.2018.04.123`.

[4] Hoda ElMaraghy, Laszlo Monostori, Guenther Schuh, and Waguih ElMaraghy. “Evolution and future of manufacturing systems”. In: *CIRP Annals* 70.2 (Jan. 1, 2021), pp. 635–658. ISSN: 0007-8506. DOI: `10.1016/j.cirp.2021.05.008`.

[5] European Commission, Directorate-General for Research and Innovation, M Breque, L De Nul, and A Petridis. *Industry 5.0 : towards a sustainable, human-centric and resilient European industry.* Publications Office, 2021. DOI: `10.2777/308407`.

[6] European Commission, Directorate-General for Research, Innovation, A Renda, S Schwaag Serger, D Tataj, A Morlet, D Isaksson, F Martins, M Mir Roca, C Hidalgo, A Huang, S Dixson-Declève, P Balland, F Bria, C Charveriat, K Dunlop, and E Giovannini. *Industry 5.0, a transformative vision for Europe : governing systemic transformations towards a sustainable industry.* Publications Office of the European Union, 2022. DOI: `doi/10.2777/17322`.

[7] European Commission, Directorate-General for Research, and Innovation. *Industry 5.0 : human-centric, sustainable and resilient.* Publications Office, 2020. DOI: `doi/10.2777/073781`.

[8] Xun Xu, Yuqian Lu, Birgit Vogel-Heuser, and Lihui Wang. “Industry 4.0 and Industry 5.0—Inception, conception and perception”. In: *Journal of Manufacturing Systems* 61 (Oct. 2021), pp. 530–535. ISSN: 02786125. DOI: `10.1016/j.jmsy.2021.10.006`.

[9] European Commission, Directorate-General for Research, Innovation, and J Müller. *Enabling Technologies for Industry 5.0 : results of a workshop with Europe’s technology leaders.* Publications Office, 2020. DOI: `doi/10.2777/082634`.

[10] Shah Zeb, Aamir Mahmood, Sunder Ali Khowaja, Kapal Dev, Syed Ali Hassan, Nawab Muhammad Faseeh Qureshi, Mikael Gidlund, and Paolo Bellavista. "Industry 5.0 is Coming: A Survey on Intelligent NextG Wireless Networks as Technological Enablers". In: (2022). Publisher: arXiv Version Number: 1. DOI: `10.48550/ARXIV.2205.09084`.

[11] Cecilia Scoccia, Marianna Ciccarelli, Giacomo Palmieri, and Massimo Callegari. *Design of a Human-Robot Collaborative System: Methodology and Case Study.* Aug. 17, 2021. DOI: `10.1115/DETC2021-70684`.

[12] Joseph E. Michaelis, Amanda Siebert-Evenstone, David Williamson Shaffer, and Bilge Mutlu. "Collaborative or Simply Uncaged? Understanding Human-Cobot Interactions in Automation". In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems.* CHI '20: CHI Conference on Human Factors in Computing Systems. Honolulu HI USA: ACM, Apr. 21, 2020, pp. 1–12. ISBN: 978-1-4503-6708-0. DOI: `10.1145/3313831.3376547`.

[13] Jéssica de Assis Dornelles, Néstor F. Ayala, and Alejandro G. Frank. "Collaborative or substitutive robots? Effects on workers' skills in manufacturing activities". In: *International Journal of Production Research* 0.0 (July 31, 2023). Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/00207543.2023.2240912, pp. 1–34. ISSN: 0020-7543. DOI: `10.1080/00207543.2023.2240912`.

[14] Alan Burden, Jared Donovan, Glenda Caldwell, and Müge Belek Fialho Teixeira. "Hybrid Digital Crafts With Collaborative Robotics". In: CAADRIA 2022: Post-Carbon. Sydney, Australia, 2022, pp. 21–30. DOI: `10.52842/conf.caadria.2022.2.021`.

[15] The Royal Danish Academy and Flemming Hansen. "The extension of the craftsman's hand by robotics". In: Nordes 2021: Matters of Scale. Aug. 15, 2021. DOI: `10.21606/nordes.2021.31`.

[16] Alessandra Papetti, Marianna Ciccarelli, Cecilia Scoccia, and Michele Germani. "A multi-criteria method to design the collaboration between humans and robots". In: *Procedia CIRP* 104 (Jan. 1, 2021), pp. 939–944. DOI: `10.1016/j.procir.2021.11.158`.

[17] Giovanni Boschetti, Maurizio Faccio, and Irene Granata. "Human-Centered Design for Productivity and Safety in Collaborative Robots Cells: A New Methodological Approach". In: *Electronics* 12 (Dec. 30, 2022), p. 167. DOI: `10.3390/electronics12010167`.

[18] C. Nuzzi, S. Pasinetti, R. Pagani, S. Ghidini, M. Beschi, G. Coffetti, and G. Sansoni. "MEGURU: a gesture-based robot program builder for Meta-Collaborative workstations". In: *Robotics and Computer-Integrated Manufacturing* 68 (2021). DOI: `10.1016/j.rcim.2020.102085`.

[19] A. Tirmizi, B.D. Cat, K. Janssen, Y. Pane, P. Leconte, and M. Witters. "User-friendly programming of flexible assembly applications with collaborative robots". In: 2019. DOI: `10.1109/REM.2019.8744135`.

[20] Federico Vicentini. “Collaborative Robotics: A Survey”. In: *Journal of Mechanical Design* 143.4 (Oct. 2020). 040802. ISSN: 1050-0472. DOI: `10.1115/1.4046238`. eprint: `https://asmedigitalcollection.asme.org/`.

[21] Andrea Casalino, Eleonora Mazzocca, Maria Grazia Di Giorgio, Andrea Maria Zanchettin, and Paolo Rocco. “Task scheduling for human-robot collaboration with uncertain duration of tasks: a fuzzy approach”. In: *2019 7th International Conference on Control, Mechatronics and Automation (ICCMA)*. 2019, pp. 90–97. DOI: `10.1109/ICCMA46720.2019.8988735`.

[22] Fei Chen, Kosuke Sekiyama, Ferdinando Cannella, and Toshio Fukuda. “Optimal Subtask Allocation for Human and Robot Collaboration Within Hybrid Assembly System”. In: *IEEE Transactions on Automation Science and Engineering* 11.4 (2014), pp. 1065–1075. DOI: `10.1109/TASE.2013.2274099`.

[23] Andrea Casalino and Angelo Geraci. “Allowing a Real Collaboration Between Humans and Robots”. In: *Special Topics in Information Technology*. Cham: Springer International Publishing, 2021, pp. 139–148. ISBN: 978-3-030-62476-7. DOI: `10.1007/978-3-030-62476-7_13`.

[24] Patrik Gustavsson, Magnus Holm, Anna Syberfeldt, and Lihui Wang. “Human-robot collaboration – towards new metrics for selection of communication technologies”. In: *Procedia CIRP* 72 (2018). 51st CIRP Conference on Manufacturing Systems, pp. 123–128. ISSN: 2212-8271. DOI: `https://doi.org/10.1016/j.procir.2018.03.156`.

[25] E. Matheson, R. Minto, E.G.G. Zampieri, M. Faccio, and G. Rosati. “Human-robot collaboration in manufacturing applications: A review”. In: *Robotics* 8.4 (2019). DOI: `10.3390/robotics8040100`.

[26] Alessandro Umbrico, Andrea Orlandini, Amedeo Cesta, Marco Faroni, Manuel Beschi, Nicola Pedrocchi, Andrea Scala, Piervincenzo Tavormina, Spyros Koukas, Andreas Zalonis, Nikos Fourtakas, Panagiotis Kotsaris, Dionisis Andronas, and S. Makris. “Design of Advanced Human–Robot Collaborative Cells for Personalized Human–Robot Collaborations”. In: *Applied Sciences* 12 (July 6, 2022), p. 6839. DOI: `10.3390/app12146839`.

[27] Shirine El Zaatari, Mohamed Marei, Weidong Li, and Zahid Usman. “Cobot programming for collaborative industrial tasks: An overview”. In: *Robotics and Autonomous Systems* 116 (June 1, 2019), pp. 162–180. ISSN: 0921-8890. DOI: `10.1016/j.robot.2019.03.003`.

[28] Lorenzo Vianello, Serena Ivaldi, Alexis Aubry, and Luka Peternel. “The effects of role transitions and adaptation in human–cobot collaboration”. In: *Journal of Intelligent Manufacturing* (Apr. 1, 2023). ISSN: 1572-8145. DOI: `10.1007/s10845-023-02104-5`.

[29] K. Castelli and H. Giberti. “A preliminary 6 Dofs robot based setup for fused deposition modeling”. In: *Mechanisms and Machine Science* 68 (2019). cited By 6, pp. 249–257. DOI: `10.1007/978-3-030-03320-0_27`.

[30] S. Pagano, R. Russo, and S. Savino. “A vision guided robotic system for flexible gluing process in the footwear industry”. In: *Robotics and Computer-Integrated Manufacturing* 65 (2020). DOI: `10.1016/j.rcim.2020.101965`.

[31] S. Pagano, R. Russo, and S. Savino. “A Smart Gluing Process by a Vision Guided Robotic System”. In: *Mechanisms and Machine Science* 91 (2021), pp. 414–422. DOI: `10.1007/978-3-030-55807-9_47`.

[32] C.-Y. Lee, T.-L. Kao, and K.-S. Wang. “Implementation of a robotic arm with 3D vision for shoes glue spraying system”. In: *ACM International Conference Proceeding Series* (2018), pp. 562–565. DOI: `10.1145/3297156.3297171`.

[33] L. Caruso, R. Russo, and S. Savino. “Microsoft Kinect V2 vision system in a manufacturing application”. In: *Robotics and Computer-Integrated Manufacturing* 48 (2017), pp. 174–181. DOI: `10.1016/j.rcim.2017.04.001`.

[34] J.-K. Oh, S. Lee, and C.-H. Lee. “Stereo vision based automation for a bin-picking solution”. In: *International Journal of Control, Automation and Systems* 10.2 (2012). cited By 42, pp. 362–373. DOI: `10.1007/s12555-012-0216-9`.

[35] E.B. Njaastad and O. Egeland. “Automatic Touch-Up of Welding Paths Using 3D Vision”. In: vol. 49. 31. cited By 13. 2016, pp. 73–78. DOI: `10.1016/j.ifacol.2016.12.164`.

[36] H. Kosler, U. Pavlovčič, M. Jezeršek, and J. Možina. “Adaptive Robotic Deburring of Die-Cast Parts with Position and Orientation Measurements Using a 3D Laser-Triangulation Sensor”. In: *Strojniski Vestnik/Journal of Mechanical Engineering* 62.4 (2016). cited By 24, pp. 207–212. DOI: `10.5545/sv-jme.2015.3227`.

[37] K. Castelli, M. Carnevale, and H. Giberti. “Development of an automatic robotic procedure for machining of skull prosthesis”. In: *Robotics* 9.4 (2020). cited By 2, pp. 1–17. DOI: `10.3390/robotics9040108`.

[38] R. Peng, D. Navarro-Alarcon, V. Wu, and W. Yang. “A Point Cloud-Based Method for Automatic Groove Detection and Trajectory Generation of Robotic Arc Welding Tasks”. In: cited By 11. 2020, pp. 380–386. DOI: `10.1109/UR49135.2020.9144861`.

[39] Jian Li, Tao Zhang, Xineng Liu, Yisheng Guan, and Danwei Wang. “A Survey of Robotic Polishing”. In: *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. Dec. 2018, pp. 2125–2132. DOI: `10.1109/ROBIO.2018.8664890`.

[40] Xi Zeng, Guangyi Zhu, Zhuohan Gao, Renquan Ji, Juwer Ansari, and Congda Lu. “Surface Polishing by Industrial Robots: A Review”. In: *The International Journal of Advanced Manufacturing Technology* (Feb. 2023). ISSN: 1433-3015. DOI: `10.1007/s00170-023-10887-3`.

[41] Simeon P. Patarinski and Roumen G. Botev. “Robot Force Control: A Review”. In: *Mechatronics* 3.4 (Aug. 1993), pp. 377–398. ISSN: 0957-4158. DOI: `10.1016/0957-4158(93)90012-Q`.

[42] Hongyi Liu and Lihui Wang. “Gesture recognition for human-robot collaboration: A review”. In: *International Journal of Industrial Ergonomics* 68 (2018), pp. 355–367. ISSN: 0169-8141. DOI: `https://doi.org/10.1016/j.ergon.2017.02.004`.

[43] M. Meghana, Ch. Usha Kumari, J. Sthuthi Priya, P. Mrinal, K. Abhinav Venkat Sai, S. Prashanth Reddy, K. Vikranth, T. Santosh Kumar, and Asisa Kumar Panigrahy. “Hand gesture recognition and voice controlled robot”. In: *Materials Today: Proceedings* 33 (2020). International Conference on Nanotechnology: Ideas, Innovation and Industries, pp. 4121–4123. ISSN: 2214-7853. DOI: `https://doi.org/10.1016/j.matpr.2020.06.553`.

[44] Xiaoling Lv, Minglu Zhang, and Hui Li. “Robot control based on voice command”. In: *2008 IEEE International Conference on Automation and Logistics*. 2008, pp. 2490–2494. DOI: `10.1109/ICAL.2008.4636587`.

[45] Patrik Gustavsson, Anna Syberfeldt, Rodney Brewster, and Lihui Wang. “Human-robot Collaboration Demonstrator Combining Speech Recognition and Haptic Control”. In: *Procedia CIRP* 63 (2017). Manufacturing Systems 4.0 – Proceedings of the 50th CIRP Conference on Manufacturing Systems, pp. 396–401. ISSN: 2212-8271. DOI: `https://doi.org/10.1016/j.procir.2017.03.126`.

[46] Dennis J. McFarland and Jonathan R. Wolpaw. “Brain-Computer Interface Operation of Robotic and Prosthetic Devices”. In: *Computer* 41.10 (2008), pp. 52–56. DOI: `10.1109/MC.2008.409`.

[47] Valeria Villani, Fabio Pini, Francesco Leali, and Cristian Secchi. “Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications”. In: *Mechatronics* 55 (2018), pp. 248–266. ISSN: 0957-4158. DOI: `https://doi.org/10.1016/j.mechatronics.2018.02.009`.

[48] L. Wang, R. Gao, J. Váncza, J. Krüger, X.V. Wang, S. Makris, and G. Chryssolouris. “Symbiotic human-robot collaborative assembly”. In: *CIRP Annals* 68.2 (2019), pp. 701–726. ISSN: 0007-8506. DOI: `https://doi.org/10.1016/j.cirp.2019.05.002`.

[49] Y. Liu, M. Habibnezhad, and H. Jebelli. “Brain-computer interface for hands-free teleoperation of construction robots”. In: *Automation in Construction* 123 (2021). DOI: `10.1016/j.autcon.2020.103523`.

[50] Rupert Ortner, Christoph Guger, Robert Prueckl, Engelbert Grünbacher, and Günter Edlinger. “SSVEP Based Brain-Computer Interface for Robot Control”. In: *Computers Helping People with Special Needs*. Ed. by Klaus Miesenberger, Joachim Klaus, Wolfgang Zagler, and Arthur Karshmer. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 85–90. ISBN: 978-3-642-14100-3. DOI: `10.1007/978-3-642-14100-3_14`.

[51] X. Chen, B. Zhao, Y. Wang, S. Xu, and X. Gao. “Control of a 7-DOF Robotic Arm System with an SSVEP-Based BCI”. In: *International Journal of Neural Systems* 28.8 (2018). DOI: `10.1142/S0129065718500181`.

[52] A. Kubacki and A. Jakubowski. “Controlling the industrial robot model with the hybrid BCI based on EOG and eye tracking”. In: vol. 2029. 2018. DOI: `10.1063/1.5066494`.

[53] Chamika Janith Perera, Isira Naotunna, Chameera Sadaruwan, R.A.R.C. Gopura, and Thilina Dulantha Lalitharatne. “SSVEP based BMI for a meal assistance robot”. In: *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. 2016, pp. 002295–002300. DOI: `10.1109/SMC.2016.7844580`.

[54] Gregory F. Rossano, Carlos Martinez, Mikael Hedelind, Steve Murphy, and Thomas A. Fuhlbrigge. “Easy robot programming concepts: An industrial perspective”. In: *2013 IEEE International Conference on Automation Science and Engineering (CASE)*. 2013, pp. 1119–1126. DOI: `10.1109/CoASE.2013.6654035`.

[55] Mohammad Amin Kuhail, Shahbano Farooq, Rawad Hammad, and Mohammed Bahja. “Characterizing Visual Programming Approaches for End-User Developers: A Systematic Review”. In: *IEEE Access* 9 (2021), pp. 14181–14202. DOI: `10.1109/ACCESS.2021.3051043`.

[56] Ahmed Magdy Ahmed Zaki, Yevheniy Dmytriyev, Kevin Castelli, Marco Carnevale, and Hermes Giberti. “Trajectory Planning for Contact-Based Robotic Applications by Use of a 3D Stereo Depth Camera”. In: *2021 3rd International Conference on Robotics and Computer Vision (ICRCV)*. 2021, pp. 68–72. DOI: `10.1109/ICRCV52986.2021.9546958`.

[57] Franz Steinmetz, Annika Wollschläger, and Roman Weitschat. “RAZER—A HRI for Visual Task-Level Programming and Intuitive Skill Parameterization”. In: *IEEE Robotics and Automation Letters* 3.3 (2018), pp. 1362–1369. DOI: `10.1109/LRA.2018.2798300`.

[58] Sonya Alexandrova, Zachary Tatlock, and Maya Cakmak. “RoboFlow: A flow-based visual programming language for mobile manipulation tasks”. In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*. 2015, pp. 5537–5544. DOI: `10.1109/ICRA.2015.7139973`.

[59] Maurizio Faccio, Irene Granata, Alberto Menini, Mattia Milanese, Chiara Rossato, Matteo Bottin, Riccardo Minto, Patrik Pluchino, Luciano Gamberini, Giovanni Boschetti, and Giulio Rosati. “Human factors in cobot era: a review of modern production systems features”. In: *Journal of Intelligent Manufacturing* 34 (May 2022), pp. 1–22. DOI: `10.1007/s10845-022-01953-w`.

[60] Hermes Giberti, Tommaso Abbattista, Marco Carnevale, Luca Giagu, and Fabio Cristini. “A Methodology for Flexible Implementation of Collaborative Robots in Smart Manufacturing Systems”. In: *Robotics* 11.1 (Feb. 2022). Number: 1 Publisher: Multidisciplinary Digital Publishing Institute, p. 9. ISSN: 2218-6581. DOI: `10.3390/robotics11010009`.

[61] Atieh Hanna, Simon Larsson, Per-Lage Götvall, and Kristofer Bengtsson. "Deliberative safety for industrial intelligent human–robot collaboration: Regulatory challenges and solutions for taking the next step towards industry 4.0". In: *Robotics and Computer-Integrated Manufacturing* 78 (Dec. 1, 2022), p. 102386. ISSN: 0736-5845. DOI: `10.1016/j.rcim.2022.102386`.

[62] *ISO 12100:2010 Safety of machinery — General principles for design — Risk assessment and risk reduction.* Standard. Geneva, CH: International Organization for Standardization, Nov. 2010.

[63] I Standard. "Iso 10218-1: 2011 robots and robotic devices—safety requirements for industrial robots—part 1: Robots". In: *International Organization for Standardization* (2011).

[64] ISO ISO. "10218-2: 2011: Robots and robotic devices–Safety requirements for industrial robots–Part 2: Robot systems and integration". In: *Geneva, Switzerland: International Organization for Standardization* 3 (2011).

[65] *ISO/TS 15066:2016 Robots and robotic devices — Collaborative robots.* Standard. Geneva, CH: International Organization for Standardization, Feb. 2016.

[66] Robin Jeanne Kirschner, Nico Mansfeld, Saeed Abdolshah, and Sami Haddadin. *ISO/TS 15066: How Different Interpretations Affect Risk Assessment.* 2022. arXiv: `2203.02706 [cs.RO]`.

[67] *Manual TM5 Series - Regular Payload Series Hardware Installation.* I623-E-02. Hardware 3.1 or earlier. Techman robots. 2020.

[68] *Manual TM12 Series - Medium & Heavy Payload Payload Series Hardware nstallation.* I624-E-05. Hardware 3.1 or earlier. Techman robots. 2023.

[69] *Onrobot 2FG7 Datasheet.* v1.4. Onrobot.

[70] *Onrobot HEX-E/H QC force/torque sensor Datasheet.* v1.1. Onrobot.

[71] *Gimatic KIT-TM-EQC20 MOUNTING INSTRUCTIONS AND OPERATING MANUAL.* Edition A. Gimatic. 2020.

[72] *Intel® RealSense Product Family D400 Series.* 337029-012. Revision 012. Intel® RealSense. Mar. 2022.

[73] David Weintrop, David C. Shepherd, Patrick Francis, and Diana Franklin. "Blockly goes to work: Block-based programming for industrial robots". In: *2017 IEEE Blocks and Beyond Workshop (B&B).* 2017, pp. 29–36. DOI: `10.1109/BLOCKS.2017.8120406`.

[74] *Software Manual TMflow.* Software Version: 1.68. Original instructions. Omron.

[75] Silvio Cocuzza, Rosanna Fornasiero, and Stefano Debei. "Novel Automated Production System for the Footwear Industry". In: *Advances in Production Management Systems. Competitive Manufacturing for Innovative Products and Services.* Ed. by Christos Emmanouilidis, Marco Taisch, and Dimitris Kiritsis. Vol. 397. Series Title: IFIP Advances in Information and Communication Technology. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 542–549. ISBN: 978-3-642-40351-4 978-3-642-40352-1. DOI: `10.1007/978-3-642-40352-1_68`.

[76] Claudio Roberto Boër and Sergio Dulio. *Mass customization and footwear: myth, salvation or reality?: a comprehensive analysis of the adoption of the mass customization paradigm in footwear, from the perspective of the EUROShoE (Extended User Oriented Shoe Enterprise) Research Project.* Springer Science & Business Media, 2007.

[77] Bojan Nemec and Leon Žlajpah. "Automation in shoe assembly". In: *Proceedings of the 15th International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD 2006), Balatonfüred, Hungary.* 2006, pp. 131–136.

[78] Bojan Nemec and Leon Zlajpah. "Shoe Grinding Cell using Virtual Mechanism Approach." In: *ICINCO-RA (1).* 2008, pp. 159–164.

[79] Z. Hu, C. Marshall, R. Bicker, and P. Taylor. "Automatic surface roughing with 3D machine vision and cooperative robot control". In: *Robotics and Autonomous Systems* 55.7 (2007), pp. 552–560. DOI: `10.1016/j.robot.2007.01.005`.

[80] Min-Gyu Kim, Juhyun Kim, Dongbin Shin, and Maolin Jin. "Robot-based Shoe Manufacturing System". In: *2018 18th International Conference on Control, Automation and Systems (ICCAS).* IEEE. 2018, pp. 1491–1494.

[81] Tae-Jung Lho, Pil-Gyu Park, Jong-Chul Suh, Dong-Joo Park, and Hee-Tae Ahn. "A Study on Development of 3D Outsole Profile Scanner for Footwear Bonding Automation". In: *Control Robot System Society: Conference Papers* (2001), pp. 131–3.

[82] G. Oliver, P. Gil, and F. Torres. "Robotic workcell for sole grasping in footwear manufacturing". In: *IEEE International Conference on Emerging Technologies and Factory Automation, ETFA* 2020-September (2020), pp. 704–710. DOI: `10.1109/ETFA46521.2020.9212058`.

[83] H.-S. Choi, G.-D. Hwang, and S.-S. You. "Development of a new buffing robot manipulator for shoes". In: *Robotica* 26.1 (2008), pp. 55–62. DOI: `10.1017/S026357470700358X`.

[84] Sergio Dulio and Claudio Boër. "Integrated production plant (IPP): an innovative laboratory for research projects in the footwear field". In: *International Journal of Computer Integrated Manufacturing* 17.7 (2004), pp. 601–611.

[85] Muhammed Selman Eryilmaz. "Analysis of shoe manufacturing factory by simulation of production processes". In: *Southeast Europe Journal of Soft Computing* 1.1 (2012). DOI: `http://dx.doi.org/10.21533/scjournal.v1i1.81`.

[86] Theodoros Staikos and Shahin Rahimifard. “An End-of-Life Decision Support Tool for Product Recovery Considerations in the Footwear Industry”. In: *Int. J. Computer Integrated Manufacturing* 20 (Sept. 2007), pp. 602–615. DOI: `10.1080/09511920701416549`.

[87] José Miguel Martín-Martínez. “Shoe Industry”. In: *Handbook of Adhesion Technology.* Cham: Springer International Publishing, 2018, pp. 1483–1532. ISBN: 978-3-319-55411-2. DOI: `10.1007/978-3-319-55411-2_51`.

[88] Giovanni Danese, Sergio Dulio, Mauro Giachero, Francesco Leporati, and Nelson Nazzicari. “A Novel Standard for Footwear Industrial Machineries”. In: *IEEE Trans. Industrial Informatics* 7 (Nov. 2011), pp. 713–722. DOI: `10.1109/TII.2011.2166789`.

[89] J.-D. Lee, C.-Y. Hsieh, Y.-J. Jhao, C.-H. Chang, C.-Y. Li, and W.-C. Li. “Implementation of automated gluing and assembly workstation”. In: *Proceedings of the 2018 IEEE International Conference on Advanced Manufacturing, ICAM 2018* (2019), pp. 276–279. DOI: `10.1109/AMCON.2018.8615008`.

[90] Gang Zhao, Guocai Ma, Wenlei Xiao, and Yu Tian. “Feature-based five-axis path planning method for robotic additive manufacturing”. In: *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 233 (Jan. 2018), p. 095440541775250. DOI: `10.1177/0954405417752508`.

[91] M. A. Kiani and H. A. Saeed. “Automatic Spot Welding Feature Recognition From STEP Data”. In: *2019 International Symposium on Recent Advances in Electrical Engineering (RAEE).* Vol. 4. 2019, pp. 1–6. DOI: `10.1109/RAEE.2019.8886989`.

[92] Francesco Jatta, Lorenzo Zanoni, Irene Fassi, and S Negri. “A roughing/cementing robotic cell for custom made shoe manufacture”. In: *International Journal of Computer Integrated Manufacturing* 17.7 (2004), pp. 645–652. DOI: `https://doi.org/10.1080/09511204200027321 2`.

[93] Xu-Zheng Liu, Jun-Hai Yong, Guo-Qin Zheng, and Jia-Guang Sun. “An offset algorithm for polyline curves”. In: *Computers in Industry* 58 (Apr. 2007), pp. 240–254. DOI: `10.1016/j.compind.2006.06.002`.

[94] Marco Carnevale, Kevin Castelli, Ahmed Magdy Ahmed Zaki, Hermes Giberti, and Cesare Reina. “Automation of Glue Deposition on Shoe Uppers by Means of Industrial Robots and Force Control”. In: *Mechanisms and Machine Science* 91 (2021), pp. 344–352. DOI: `10.1007/978-3-030-55807-9_39`.

[95] *Robots and robotic devices — Safety requirements for industrial robots — Part 1: Robots.* ISO 10218-1:2011.

[96] Zhigang Yang, Yi An, Yanbin Sun, and Jingchun Zhang. “Research on intelligent glue-coating robot based on visual servo”. In: *Physics Procedia* 24 (2012), pp. 2165–2171. DOI: `doi.org/10.1016/j.phpro.2012.02.318`.

[97] Iñaki Maurtua, Aitor Ibarguren, and Alberto Tellaeche. "Robotic solutions for footwear industry". In: *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012)*. IEEE. 2012, pp. 1–4. DOI: `https://doi.org/10.1109/ETFA.2012.6489780`.

[98] Iñaki Maurtua, Aitor Ibarguren, and Alberto Tellaeche. "Robotics for the benefit of footwear industry". In: *International Conference on Intelligent Robotics and Applications*. Springer. 2012, pp. 235–244. DOI: `https://doi.org/10.1007/978-3-642-33515-0_24`.

[99] *Executive Summary World Robotics 2021 Industrial Robots*. `https://ifr.org/img/worldrobotics/Executive_Summary_WR_Industrial_Robots_2021.pdf`. Lyoner Str. 18 Frankfurt am Main, 2021.

[100] Kevin Castelli, Ahmed Magdy Ahmed Zaki, Yevheniy Dmytriyev, Marco Carnevale, and Hermes Giberti. "A Feasibility Study of a Robotic Approach for the Gluing Process in the Footwear Industry". In: *Robotics* 10.1 (2021). ISSN: 2218-6581. DOI: `10.3390/robotics10010006`.

[101] Khalida Douibi, Solène Le Bars, Alice Lemontey, Lipsa Nag, Rodrigo Balp, and Gabrièle Breda. "Toward EEG-Based BCI Applications for Industry 4.0: Challenges and Possible Applications". In: *Frontiers in Human Neuroscience* 15 (2021). ISSN: 1662-5161. DOI: `10.3389/fnhum.2021.705064`.

[102] Anthony M. Norcia, L. Gregory Appelbaum, Justin M. Ales, Benoit R. Cottereau, and Bruno Rossion. "The steady-state visual evoked potential in vision research: A review". In: *Journal of Vision* 15.6 (May 2015), pp. 4–4. ISSN: 1534-7362. DOI: `10.1167/15.6.4`. eprint: `https://arvojournals.org/arvo/content\_public/journal/jov/933934/i1534-7362-15-6-4.pdf`.

[103] Aravind Ravi, Nargess Heydari, and Ning Jiang. "User-Independent SSVEP BCI Using Complex FFT Features and CNN Classification". In: *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*. 2019, pp. 4175–4180. DOI: `10.1109/SMC.2019.8914258`.

[104] Fabrizio Beverina, Giorgio Palmas, Stefano Silvoni, Francesco Piccione, Silvio Giove, et al. "User adaptive BCIs: SSVEP and P300 based interfaces." In: *PsychNology J.* 1.4 (2003), pp. 331–354.

[105] G. H. Klem, H. O. Lüders, H. H. Jasper, and C. Elger. "The ten-twenty electrode system of the International Federation. The International Federation of Clinical Neurophysiology". In: *Electroencephalogr Clin Neurophysiol Suppl* 52 (1999), pp. 3–6.

[106] Hermann Hinrichs, Michael Scholz, Anne Katrin Baum, Julia W. Y. Kam, Robert T. Knight, and Hans-Jochen Heinze. "Comparison between a wireless dry electrode EEG system with a conventional wired wet electrode EEG system for clinical applications". In: *Scientific Reports* 10.1 (Mar. 2020), p. 5218. ISSN: 2045-2322. DOI: `10.1038/s41598-020-62154-0`.

[107] Xiao Xing, Yijun Wang, Weihua Pei, Xuhong Guo, Zhiduo Liu, Fei Wang, Gege Ming, Hongze Zhao, Qiang Gui, and Hongda Chen. “A High-Speed SSVEP-Based BCI Using Dry EEG Electrodes”. In: *Scientific Reports* 8.1 (Oct. 2018), p. 14708. ISSN: 2045-2322. DOI: `10.1038/s41598-018-32283-8`.

[108] Omid G. Sani. *Quick SSVEP: A web-based SSVEP stimulation interface.* Version 2.0. Aug. 2020. DOI: `10.5281/zenodo.3992582`.

[109] Jonathan Peirce. “Generating stimuli for neuroscience using PsychoPy”. In: *Frontiers in Neuroinformatics* 2 (2009), p. 10. ISSN: 1662-5196. DOI: `10.3389/neuro.11.010.2008`.

[110] I. Volosyak, H. Cecotti, and A. Gräser. “Steady-state visual evoked potential response - Impact of the time segment length”. In: vol. 1. 2010, pp. 284–288.

[111] Yu Ogura, Masakazu Fujii, Kazuyuki Nishijima, Hiroki Murakami, and Mitsuharu Sonehara. “Applicability of Hand-Guided Robot for Assembly-Line Work”. In: *Journal of Robotics and Mechatronics* 24.3 (2012), pp. 547–552. DOI: `10.20965/jrm.2012.p0547`.

[112] J. Krüger, T.K. Lien, and A. Verl. “Cooperation of human and machines in assembly lines”. In: *CIRP Annals* 58.2 (2009), pp. 628–646. ISSN: 0007-8506. DOI: `https://doi.org/10.1016/j.cirp.2009.09.009`.

[113] A. Liu, Q. Liu, X. Zhang, X. Chen, and X. Chen. “Muscle Artifact Removal Toward Mobile SSVEP-Based BCI: A Comparative Study”. In: *IEEE Transactions on Instrumentation and Measurement* 70 (2021). DOI: `10.1109/TIM.2021.3085944`.

[114] Mohammad Safeea, Richard Bearee, and Pedro Neto. “End-Effector Precise Hand-Guiding for Collaborative Robots”. In: *ROBOT 2017: Third Iberian Robotics Conference.* Ed. by Anibal Ollero, Alberto Sanfeliu, Luis Montano, Nuno Lau, and Carlos Cardeira. Cham: Springer International Publishing, 2018, pp. 595–605. ISBN: 978-3-319-70836-2.

[115] OnRobot. *Case Study: Hand Guiding with OnRobot Force Torque Sensor.* `https://onrobot.com/en/case-studies/hand-guiding-with-onrobot-force-torque-sensor`. Odense SØ, Denmark.

[116] Christina Schmidbauer, Sebastian Schlund, Tudor B Ionescu, and Bernd Hader. “Adaptive task sharing in human-robot interaction in assembly”. In: *2020 IEEE international conference on industrial engineering and engineering management (IEEM).* IEEE. 2020, pp. 546–550.

[117] Jeffrey Too Chuan Tan, Feng Duan, Ye Zhang, and Tamio Arai. “Task decomposition of cell production assembly operation for man-machine collaboration by HTA”. In: *2008 IEEE International Conference on Automation and Logistics.* 2008, pp. 1066–1071. DOI: `10.1109/ICAL.2008.4636309`.

[118] Yubai Wang, Hao Wang, Yulin Zhou, Zhen Zhang, Qiang Zhou, Changkun Diao, Nan Su, and Jianwei Niu. “Hierarchical Task Analysis of Human-Robot Collaboration Scenario in Space Station”. In: *2022 IEEE 17th Conference on Industrial Electronics and Applications (ICIEA).* IEEE. 2022, pp. 1640–1645.

[119] Neville A Stanton. “Hierarchical task analysis: Developments, applications, and extensions”. In: *Applied ergonomics* 37.1 (2006), pp. 55–79.

[120] John Patrick, A Gregov, and P Halliday. “Analysing and training task analysis”. In: *Instructional science* 28 (2000), pp. 51–79.

[121] Dario Antonelli, Sergey Astanin, and Giulia Bruno. “Applicability of Human-Robot Collaboration to Small Batch Production”. In: *Collaboration in a Hyperconnected World*. Ed. by Hamideh Afsarmanesh, Luis M. Camarinha-Matos, and António Lucas Soares. Cham: Springer International Publishing, 2016, pp. 24–32. ISBN: 978-3-319-45390-3.

[122] Christoph Petzoldt, Matthias Harms, and Michael Freitag. “Review of task allocation for human-robot collaboration in assembly”. In: *International Journal of Computer Integrated Manufacturing* (2023), pp. 1–41.

[123] Rainer Müller, Matthias Vette, and Ortwin Mailahn. “Process-oriented Task Assignment for Assembly Processes with Human-robot Interaction”. In: *Procedia CIRP* 44 (2016). 6th CIRP Conference on Assembly Technologies and Systems (CATS), pp. 210–215. ISSN: 2212-8271. DOI: `https://doi.org/10.1016/j.procir.2016.02.080`.

[124] Giulia Bruno and Dario Antonelli. “Dynamic task classification and assignment for the management of human-robot collaborative teams in workcells”. In: *The International Journal of Advanced Manufacturing Technology* 98 (2018), pp. 2415–2427.

[125] Jeffrey Too Chuan Tan, Feng Duan, Ryu Kato, Tamio Arai, and Ernest Hall. *Collaboration planning by task analysis in human-robot collaborative manufacturing system*. INTECH Open Access Publisher, 2010.

[126] Yee Yeng Liau and Kwangyeol Ryu. “Genetic algorithm-based task allocation in multiple modes of human–robot collaboration systems with two cobots”. In: *The International Journal of Advanced Manufacturing Technology* 119.11-12 (2022), pp. 7291–7309.

[127] Alessandro Alessio, Khurshid Aliev, and Dario Antonelli. “Multicriteria task classification in human-robot collaborative assembly through fuzzy inference”. In: *Journal of Intelligent Manufacturing* (2022), pp. 1–19.

[128] Luca Gualtieri, Erwin Rauch, Renato Vidoni, and Dominik T. Matt. “An evaluation methodology for the conversion of manual assembly systems into human-robot collaborative workcells”. In: *Procedia Manufacturing* 38 (2019). 29th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM 2019), June 24-28, 2019, Limerick, Ireland, Beyond Industry 4.0: Industrial Advances, Engineering Education and Intelligent Manufacturing, pp. 358–366. ISSN: 2351-9789. DOI: `https://doi.org/10.1016/j.promfg.2020.01.046`.

[129] Ali Ahmad Malik and Arne Bilberg. “Complexity-based task allocation in human-robot collaborative assembly”. In: *Industrial Robot: the international journal of robotics research and application* (2019).

[130] George Evangelou, Nikos Dimitropoulos, George Michalos, and Sotiris Makris. "An approach for task and action planning in human–robot collaborative cells using AI". In: *Procedia Cirp* 97 (2021), pp. 476–481.

[131] Flor A. Bravo, Alejandra M. González, and Enrique González. "A review of intuitive robot programming environments for educational purposes". In: *2017 IEEE 3rd Colombian Conference on Automatic Control (CCAC)*. 2017, pp. 1–6. DOI: `10.1109/CCAC.2017.8276396`.

[132] Iina Aaltonen and Timo Salmi. "Experiences and expectations of collaborative robots in industry and academia: barriers and development needs". In: *Procedia Manufacturing* 38 (2019). 29th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM 2019), June 24-28, 2019, Limerick, Ireland, Beyond Industry 4.0: Industrial Advances, Engineering Education and Intelligent Manufacturing, pp. 1151–1158. ISSN: 2351-9789. DOI: `https://doi.org/10.1016/j.promfg.2020.01.204`.

[133] Ana M Djuric, RJ Urbanic, and JL Rickli. "A framework for collaborative robot (CoBot) integration in advanced manufacturing systems". In: *SAE International Journal of Materials and Manufacturing* 9.2 (2016), pp. 457–464.

[134] Gopika Ajaykumar, Maureen Steele, and Chien-Ming Huang. "A Survey on End-User Robot Programming". In: *ACM Comput. Surv.* 54.8 (Oct. 2021). ISSN: 0360-0300. DOI: `10.1145/3466819`.

[135] Christoph Mayr-Dorn, Mario Winterer, Christian Salomon, Doris Hohensinger, and Rudolf Ramler. "Considerations for using Block-Based Languages for Industrial Robot Programming - a Case Study". In: *2021 IEEE/ACM 3rd International Workshop on Robotics Software Engineering (RoSE)*. 2021, pp. 5–12. DOI: `10.1109/RoSE52553.2021.00008`.

[136] Daniela Fogli, Luigi Gargioni, Giovanni Guida, and Fabio Tampalini. "A hybrid approach to user-oriented programming of collaborative robots". In: *Robotics and Computer-Integrated Manufacturing* 73 (2022), p. 102234. ISSN: 0736-5845. DOI: `https://doi.org/10.1016/j.rcim.2021.102234`.

[137] Andreas Hansen, Valeria Villani, Andrea Pupa, and Astrid Heidemann Lassen. *Introducing novice operators to collaborative robots: a hands-on approach for learning and training.* May 12, 2023. DOI: `10.36227/techrxiv.22762562.v1`.

[138] Yevheniy Dmytriyev, Federico Insero, Marco Carnevale, and Hermes Giberti. "Brain–Computer Interface and Hand-Guiding Control in a Human–Robot Collaborative Assembly Task". In: *Machines* 10.8 (Aug. 5, 2022), p. 654. ISSN: 2075-1702. DOI: `10.3390/machines10080654`.

[139] Yevheniy Dmytriyev, Marco Carnevale, Hermes Giberti, and Giovanni Todeschini. "On cobot programming in industrial tasks: a test case". In: *2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*. 2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA). Ankara, Turkey: IEEE, June 9, 2022, pp. 1–9. ISBN: 978-1-66546-835-0. DOI: `10.1109/HORA55278.2022.9800018`.