# Enhanced Cognition for Adaptive Human-Robot Collaboration

Alessandro Umbrico*, Mikel Anasagasti†, Stefan-Octavian Bezrucav‡, Francesca Canale§, Amedeo Cesta*, Burkhard Corves‡, Nils Mandischer‡, Mikel Mondragon†, Cristina Naso Rappis §, Andrea Orlandini*

* Institute of Cognitive Science and Technologies, National Research Council, Italy.
{name.surname}@istc.cnr.it
† GOIZPER Group, Spain.
{manasagasti,mmondragon}@goizper.com
‡ RWTH Aachen University, Institute of Mechanims Theory, Machine Dynamics and Robotics, Germany.
{bezrucav,corves,mandischer}@igmr.rwth-aachen.de
§ STAM, Italy.
{f.canale,c.nasorappis}@stamtech.com

***Abstract*—Cyber-Physical Systems constitute one of the core concepts in Industry 4.0 aiming at realizing production systems that combine the efforts of human workers, robots, and intelligent entities. This is particularly crucial in Human-Robot Collaboration manufacturing where a tight peer-to-peer interaction between humans and intelligent autonomous robots is necessary. The work proposes the integration of novel Artificial Intelligence technologies to enhance the flexibility and adaptability of collaborative robots. The integrated functionalities allow a collaborative robot to autonomously recognize the tasks a human worker performs, and accordingly adapt its behavior. The approach is deployed on a real HRC scenario showing the functioning of the developed cognitive capabilities and the increased flexibility of resulting collaborations.**

***Index Terms*—Human-Robot Collaboration, Robot Perception, Knowledge Representation and Reasoning, Automated Planning**

## I. Introduction

Cyber-Physical Systems (CPS) constitute one of the core concepts in Industry 4.0 whose aim is to push manufacturing systems towards an ever tighter integration of computational capabilities with physical processes and entities [1]. Considering an *antropo-centric* evolution of CPSs [2], research should push coexistence and combination of skills of humans and automation [3]. Modern and future CPSs should increasingly take into account the *human factor* which is crucial to design efficient, safe and acceptable synergetic interactions between human and artificial actors [4]. This is especially true in Human-Robot Collaboration (HRC) scenarios where higher levels of *flexibility* and *adaptability* are necessary to dynamically adapt *control policies* to production needs as well as different *skills* and *features* of workers that take part in collaborative processes [5], [6].

A symbiotic coexistence of human operators and autonomous robots raises several technological challenges since human behaviors are neither predictable nor controllable. Classical control processes that usually rely on static models of robot capabilities and production dynamics are today obsolete [7]. Robot controllers require higher levels of flexibility and adaptability to effectively support dynamic production environments [8]. Human-Robot Interaction scenarios are intrinsically characterized by *uncertainty* due to the coexistence of *uncontrollable* human agents and *controllable* artificial agents [9]. Artificial Intelligence (AI) can play a role in endowing controllers and thus robots with the *cognitive skills* needed to effectively and reliably behave in such scenarios [10], [11]. Towards this goal, we are investigating the use of novel AI technologies in HRC to allow collaborative robots to: (i) *perceive* the environment [12], correctly *interpret* occurring events and situations to build and maintain *knowledge* about production contexts [13]; (ii) reason about their own *capabilities/skills* and dynamically contextualize possible actions according to the *known* state of a production scenario [14], [15] and; (iii) autonomously decide *how* to *interact* with the environment and other "actors" (i.e., human operators but even other robots if necessary) in order to carry out tasks and dynamically support production needs [16]–[18].

This paper presents an original integration of technologies realized within a EU-funded research project, Sharework [1]. It describes the integration of novel AI-based perception, representation and planning modules and their deployment in a real-world HRC scenario. The main contribution is the integration of AI-based modules that support the synthesis of flexible behaviors by enhancing *cognitive skills*, *decisional autonomy* and *awareness* of collaborative robots. A *perception module* allows the robot to autonomously recognize objectives of the environment and *states/situations* that are relevant with respect to the production. A *semantic module* allows the robot to internally represent production processes and reason about tasks the worker and the robot itself could perform to achieve desired objectives. A *planning module* allows the robot to au-

This work is partially supported by ShareWork project (H2020, European Commission – G.A. 820807).

[1]https://sharework-project.eu

tonomously synthesize and execute actions that are necessary to support collaborative processes, according to observations and "triggers" received from the environment. Such generic HRC control system is then deployed and demonstrated to be effective in dealing with a realistic manufacturing scenario.

Section II provides a brief overview of the Sharework project with a general description of the functionalities supported by the developed modules. Section III describes the real-world HRC scenario considered in this work, and the involved modules. It specifically describes with more details the technologies behind these modules and their "tailoring" to the needs of the scenario. Section IV describes the integration scheme and near-real-time deployment of modules in the real-world scenario. This section aims at showing the increased flexibility, awareness and autonomy provided to the robot through the integration of the developed modules and underlying AI technologies. Section V then concludes the paper summarizing the contribution and discussing some improvements and future developments.

## II. THE SHAREWORK PROJECT

The Sharework project aims at developing an effective and safe CPS system for anthropocentric *Human Robot Collaboration* with no fences. It defines a novel architecture made of 15 software and hardware modules, supporting *modularity* through a wide range of *configurations* in order to *customize* the deployment according to the different and specific industrial needs. This work specifically focuses on the integration of the modules responsible for enhancing *perception* and *decisional* capabilities of HRC cells.

### A. Environment Cognition

The environment cognition module of Sharework combines high-level and explainable AI (XAI), and artificial cognition algorithms to enable any factory personnel to implement cognition applications without knowledge about machine learning in general. The module consists of multiple sub-modules applicable to different industrial domains and application scenarios: (i) To supervise and allocate tasks on a widespread work space, typical workshop items are localized and classified with unsupervised segmentation and deep learning techniques [12]. Training the neural networks is made lenient, as training data is directly generated from the segmentation data, and may therefore be applied with low effort on the work cell. Typical items are screw drivers, wrenches, drills, or hammers. (ii) To supervise collaborative screwing processes, another sub-module allows to locate screws inserted into a work piece. By applying XAI and an intuitive user interface, the screw detection can be commissioned within five minutes[2]. XAI means that the user may understand how the algorithm comes to its conclusion. Therefore, we define a total five simple features arranged in a Random Forest classifier [19].

All sub-modules are implemented in a cognition framework [12] that, on the one hand, deals with accelerating computation by applying efficient data structures such that all methodology is available in real- or online-time depending on the use-case. Fast computation is an inherent feature required to allow further computation based on cognition data. On the other hand, high-level AI and machine learning techniques are wrapped for more lenient usage, which makes it easier for untrained or personnel less knowledgeable in the methodical background.

### B. Hybrid Reasoning for Behavior Synthesis

The knowledge base module and the task planning module of the Sharework architecture support *hybrid reasoning* capabilities enhancing *autonomy*, *flexibility* and *context awareness* of collaborative robots [17], [20].

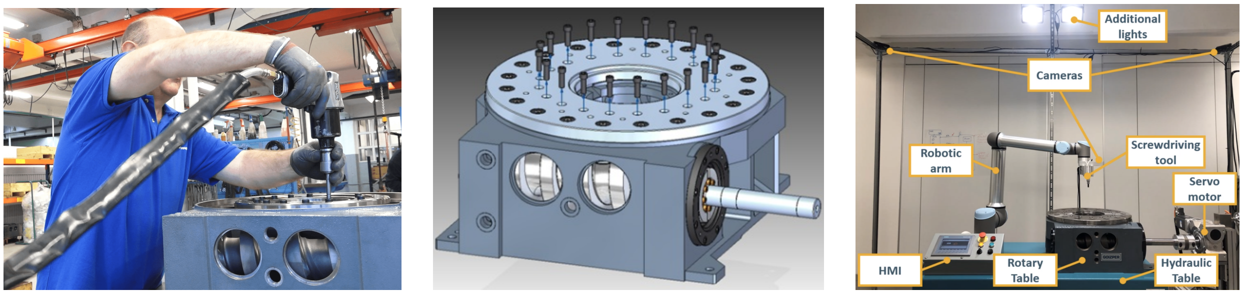
The knowledge base serves as a *centralized repository* aggregating and "abstracting" information gathered from other modules to *infer* the current state of a HRC cell and production processes. It interprets and refines *knowledge* about a collaborative production scenario using a *domain ontology* [21] developed within Sharework, called SOHO [13]. This ontology has been designed to characterize collaborative processes according to three main synergetic perspectives, formalized through contexts: (i) *environment context*; (ii) *behavior context* and; (iii) *production context*. The task planning module encapsulate *automated planning and execution capabilities* [22], [23] to dynamically synthesize robot and expected worker behaviors and *reliably* carry out production processes. This module specifically relies on the timeline-based planning formalism [24] and coordinates human and robot skills through the synthesis of synchronized *time-aware* and *controllability-aware* [25] behaviors (i.e., *timelines*) that take into account both *safety* and *efficiency* of a collaborative process [18].

The tight integration of these two modules support higher levels of flexibility and adaptability of the whole HRC cell [14]. On the one hand it supports *dynamic reconfiguration* of planning models according to the *known* physical configuration of a HRC cells and skills/features of the "acting agents" that take part to collaborative processes [5], [14], [15]. On the other hand, it supports *online detection* of relevant *situations* and *events* triggering automatic adaptation of robot behaviors.

## III. THE GOIZPER USE CASE

The HRC scenario subject of this work takes into account the shop-floor of a company offering differential and global solutions in power transmission and spraying components. It specifically considers a servo rotary table that is assembled in seven fixed assembly stations. In each station there is an operator performing a specific task of the assembly process. All tasks are carried out manually, just using cranes and lifters to transport the heavy components from one station to another. The collaboration between the human operator and the robot concerns two out of seven tasks of the rotary table assembly process. Figure 1 shows part of the physical environment (Figure 1(a)) and the workpiece (the rotary table) considered in the current work (Figure 1(b)).

[2]https://www.youtube.com/watch?v=vNPjjKpIDWw




(a) (b) (c)

Fig. 1. Pictures (a) and (b) show the original scenario and the target workpiece. Picture (c) shows the scenario after the introduction of the *cobot*.

TABLE I
ROTARY TABLE BOLTS TIGHTENING STEP BY STEP TASK EXECUTION SEQUENCE

| Step | Operator | Robot |
|---|---|---|
| 1 | Place the rotary table on the working station | Rest position |
| 2 | Mount correct bit on the tightening tool | |
| 3 | Select the robot program on the HMI | |
| 4 | Put adhesive on bolts and insert them in the rotary table holes | Identify inserted bolts |
| 5 | | Screw each bolt with required torque |
| 6 | | Signal correct operations/errors to operator through the HMI |

The introduction of collaborative robots in this scenario aims to improve the working condition of operators by supporting them in their most low added value and repetitive tasks and in handling heavy parts. We thus specifically consider the bolt tightening and torque measuring task, which in particular is repetitive and alienating. Moreover, the ergonomic of the operator is not always optimal when performing the manual task, due to the wide area of the working station that needs to be covered.

Figure 1(c) shows the obtained physical environment where the working unit is equipped with a collaborative robot (an UR10e robotic arm) integrated with an electric screwdriving tool which allows to control the screwing torque. RGBD cameras monitor the working environment, to ensure a safe HRC, and one additional camera is mounted on the robotic arm to have a clear view of the rotary table. Broadly speaking, the process requires the human and the robot to "collaboratively" perform a number ($N = 16$) of screwing operations of bolts on the rotary table of Figure 1(b) (i.e., the workpiece). Bolts are supposed to be covered with adhesive and then placed by the worker on the rotary table while the robot should dynamically "react" accordingly, by screwing those that are "ready" with a required torque value (69 Nm). Table I provides the detailed workflow of the assembly task performed collaboratively between the human operator and the robot, where tasks executed in parallel are indicated in the same row.

In this specific scenario, collaborative robotics is preferable to a fully automated solution. This is because the presence of the human operator is necessary to ensure the flexibility of the process. A fully automated solution would have been possible (for example implementing cooperation between two robots, one that places the bolts and the second that screws them), but this would have been a robotic process tailored to a specific product, losing the level of flexibility required by the end user. Moreover, a fully automated solution would have required a significant increase in costs (having to purchase many duplicate components, such as robot and tool), and a much higher degree of implementation complexity, having to take care, for example, of the coordination of movement of two different robots. Since the task of placing the screws in the rotary table is not heavy or ergonomically dangerous for operators, the implementation of such a complex solution was not considered justified and the collaborative solution proposed in this document was preferred. In addition, the operator plays an important role of supervisor of the whole process, by continuously monitoring that the screwdriving tool applies the required torque to tighten the bolts, role that could not be substituted by an automatic process.

The key aspects of this scenario are the *perception capabilities* required to automatically *recognize* the correct position of the bolts and *synchronization capabilities* needed to recognize *opportunities of actions* for the robot and synthesize robot behaviors (i.e., screwing of bolts).

### A. Deployment of Perception Capabilities

The main objective of the perception module is to detect screw candidates and classify them into holes and screws. Hence, the robot gains knowledge on where screwing actions may be performed. For input, a camera is mounted concentrically over the rotary disk, facing the main surface area (see Figure 1(b)). Screw detection is based on tree-learning [19] with a small number of features to maintain a certain level of understandability [12]. The perception module is capable of finding silver and black screws within varying lighting conditions. To train the classifier, the user is consulted with a GUI. First, images are taken using the camera. Then, screw candidates are segmented using an underlying Hough Circle

Transform [26], and then presented to the user. The user then decides whether the candidate is a screw, hole, or an erroneous segmentation by pressing the according buttons. Using only about 60 samples the classifier is trained successfully and may be directly deployed in the application. Depending on the lighting conditions an accuracy measure between 85.3% and 94.7% is realized which is fully sufficient to perform autonomous screwing in the use-case.

### *B. Production and Planning Knowledge*

To achieve flexible and proactive production-level support, the knowledge base encapsulates a model of production dynamics and tasks the human and the robot should perform. The deployed knowledge base instantiates the general model of the SOHO ontology [13] to contextualize production events and human/robot skills. Production tasks characterizing "bolt screwing" are in this case represented as *synchronous* collaborative operations [27], [28] (i.e., instances of the ontological class `SynchronousHRCTask`). Each screwing operation indeed requires the human and robot working on the same target/workpiece (i.e., a specific part composing the whole rotary table), following a strict *temporal ordering* in the execution of their operations. The robot should start screwing a bolt only after the worker has placed it on a hole of the rotary table.

Considering the *Taxonomy of Functions* [29] integrated in SOHO, each *screwing task* is decomposed into two *functions*. As instance of `SynchronousHRCTask`, a *human function* is executed before a *robot function* and they are respectively: (i) a `PickPlace` function performed by the worker to place a bolt on the workpiece; (ii) a `Screw` function of the robot to actually tighten placed bolts. The production procedure of the GOIZPER use case is encapsulated into the knowledge base in shape of a *decomposition graph*. This graph describes the hierarchical decomposition of high-level production tasks (i.e., goals) into increasingly simpler tasks that can be performed by the human and/or by the robot. Specifically, the high-level `ProductionGoal rotary-table-assembly` is decomposed into a number of (complex) `ConjunctiveTask`, one for each *screwing task* that can be performed. The leaves of the graph thus are the *functions* (i.e., low-level primitive operations) the human and the robot should perform.

The knowledge is used to configure the task planning module [14] and automatically trigger *production goals* according to detected events and current state of production. Production goals entail either the execution of the whole process or part of it like e.g., the execution of a single *screwing task* of a specific bolt of the rotary table. In the considered scenario, production goals concern single screwing tasks in order to synthesize robot behaviors adapted to the observed behavior of the worker. In particular, the planner further decomposes `Screw` functions of the robot into the primitive *motions* and *tool commands* (activation and deactivation of the screwdriver) necessary to actually implement them and dynamically support production. The developed task planner module encapsulates the open-source framework PLATINUm [3] [30] and integrates timeline-based planning and execution capabilities into ROS through ROXANNE [4].

## IV. Adaptive Collaboration through Layered Perception and Reasoning

The environment cognition, the knowledge base and the task planner modules have been integrated into a ROS-based control architecture in order to support the collaborative production scenario depicted in Figure 1(c) [5]. Figure 2 shows the integration scheme of the developed modules. The modules can be organized into two architectural levels. An *environment-aware level* encapsulates ROS modules that directly interact with the production environment abstracting data and executing skills. A *production-aware level* encapsulates ROS modules that contextualize detected information and synthesize robot behaviors by taking into account production needs and requirements of the considered scenario.

First the knowledge base module is *initialized* building a *knowledge graph* based on standard semantic technologies (OWL [31]). The knowledge base completely characterizes the production scenario providing the whole architecture with a semantic representation compliant with the SOHO ontology [13]. The knowledge base then is ready to *interpret* data received from the environment cognition module and *trigger* suitable production goals to the task planner module. The timeline-based model of the task planner is automatically generated from the knowledge base in order to operationally support *known* production goals. Every time a goal is triggered by the knowledge base, the task planner synthesizes plans whose execution coordinates low-level skills of the robot issuing suitable commands. As Figure 2 shows, the task planner in this case dispatches commands to ROS nodes "exposing" a functional interface for the robotic arm (i.e., the motion planner) and the screwdriver (i.e., the screwdriver controller). Execution *feedback* notifies the task planner about the correct execution of issued commands or *failures*. Table II shows with more details the actual topic and messages used to exchange information between modules and to implement the design cognitive control flow.

The scenario has been designed as "reactive" where the robot uses *perception capabilities* to *autonomously react* to the observed behavior of the worker, deciding the *task* to perform in order to support the collaborative process. The worker behaves freely and applies the adhesive and places one or more bolts to be screwed on the work piece according to her/his preferences. The environment cognition's processing actions are performed on the pictures acquired by a camera that is mounted directly on the robotic arm, ensuring that its field of view includes the whole surface of the rotary table. The example of an image taken by the camera and processed by the environment cognition module is reported in Figure 3(a),

[3]https://github.com/pstlab/PLATINUm.git

[4]https://github.com/pstlab/roxanne_rosjava.git

[5]All Sharework modules are compliant with ROS Melodic http://wiki.ros.org/melodic

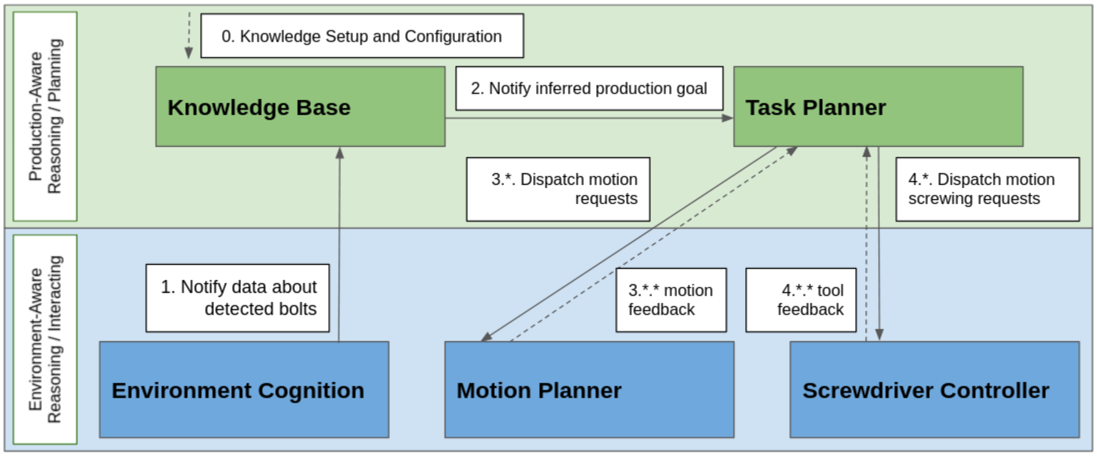



Fig. 2. Architectural view of the integrated Sharework modules. The knowledge base and the task planner are configured with a model of the GOIZPER scenario. Every time the environment cognition detects some bolt to be screwed on the rotary table, it notifies the knowledge base. The knowledge base contextualizes received data with respect to the know production procedure and triggers a production goal to the task planner. The task planner then synthesizes and executes a suitable robot behavior by dispatching commands to the motion planner and the screwdriver controller.

TABLE II
ROS-BASED INTEGRATION DETAILS SHOWING TOPICS AND MESSAGES USED TO IMPLEMENT THE "CONTROL FLOW" OF FIGURE 2

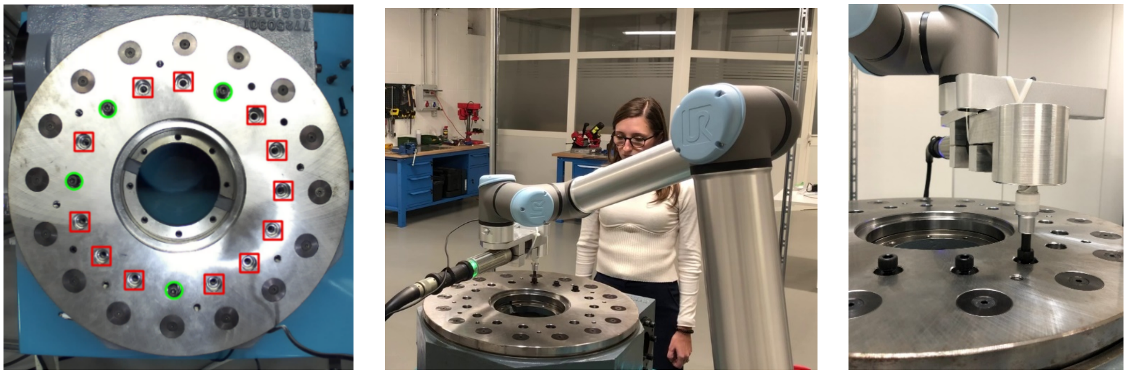
| ROS Node | Node Input | Node Output |
|---|---|---|
| Environment Cognition | - | topic: /screw_detector/screw_to_be_tightened<br>msg: sharework_cognition_msgs/DetectionResult |
| Knowledge Base | topic: /screw_detector/screw_to_be_tightened<br>msg: sharework_cognition_msgs/DetectionResult | topic: /sharework/taskplanner/request<br>msg: task_planner_interface_msgs/TaskPlanningRequest |
| Task Planner | topic: /sharework/taskplanner/request<br>msg: task_planner_interface_msgs/TaskPlanningRequest | topic: /cartesian_pose<br>msg: geometry_msgs/Pose |
| | | topic: /sharework/taskplanner/goal<br>msg: task_planner_interface_msgs/TaskExecutionRequest |
| Motion Planner (*) | topic: /cartesian_pose<br>msg: geometry_msgs/Pose | topic: /motion_result<br>msg: task_planner_interface_msgs/MotionResult |
| ScrewDriver Controller (*) | topic: /sharework/taskplanner/goal<br>msg: task_planner_interface_msgs/TaskExecutionRequest | topic: /sharework/taskplanner/feedback/robot<br>msg: task_planner_interface_msgs/TaskExecutionFeedback |

(*) Messages on output topics are used by the the task planner as execution feedback of plans.

where the bolts detected by the algorithm on the surface of the rotary table are marked with a green circle, and the holes with a red square. In addition to autonomously recognizing the bolt placement, the environment cognition module allows the HRC cell to detect the physical location of the bolts placed by the worker by extracting related geometric coordinates expressed with respect to the camera frame.

Before the coordinates of the detected bolts are dispatched to the knowledge base, they must be converted into the robot reference system. The algorithm that performs the conversion calculates the distance between the detected bolt and the center of the rotary table in the camera frame and then converts it into the robot frame. This solution was feasible since, as mentioned above, the camera used to acquire pictures of the rotary table was mounted on the robotic arm, making its absolute position and orientation easily controllable. After the conversion is performed for each detected screw, the translated positions are dispatched to the knowledge base module which contextualizes received data with respect to the known production procedures and operational constraints.

This is done by taking into account the structure of *known* production processes of the scenario but also the current state of production. This means that for example, the knowledge base knows bolts that have been screwed already and therefore it would be able to *validate* data received from the environment. Namely, the knowledge base would automatically recognize *false positives* (i.e., notifications about bolts already screwed) and avoid unnecessary operations. If the internal validation of the knowledge base is successful, it asynchronously triggers a *planning goal* signal to the task planning module in order to synthesize robot tasks suitable to dynamically support production. In this case, the task planner module receives a goal specifying a list of bolts placed by the worker and detected by the environment cognition module. The planner synthesizes robot motions and screwing actions that are online dispatched to the robot controller for their actual execution (Figure 3(b) and Figure 3(c)) following the scheme of Figure 2 and Table II. In order to verify if the screwing action was successful (i.e., the correct torque value was reached during the tightening) the corresponding output of the electric screwdriver is checked and sent as feedback to the task planner.



(a) (b) (c)

Fig. 3. Automatic execution of the bolt tightening and torque measuring task. The first picture (a) shows the automatic screw detection performed by the perception module; the second two pictures (b and c) show the actual tightening process of one screw.

## V. Final Remarks and Future Works

This work shows the integration of perception, representation and reasoning capabilities into a ROS-based cognitive control architecture to enhance flexibility and awareness of HRC cells. Specifically the paper shows the deployment of AI-based modules developed with the EU H2020 project Sharework on a real-world production scenario. These modules effectively support advanced cognitive control capabilities allowing a HRC cell to: (i) autonomously detect and abstract relevant events from the environment (i.e. placement of bolts); (ii) contextualize gathered information and recognize *opportunity of actions* according to *known* production needs, and; (iii) synthesize suitable *production tasks* (i.e. *bolt screwing tasks*) and coordinate robotic skills to implement the needed low-level operations (i.e. robot motions and activation/deactivation of the screwdriver). Future works will further generalize the developed architecture by taking into account different scenarios. They will consider also the integration of cognitive capabilities necessary to evaluate *safety constraints* of a HRC cell at different levels of abstraction and accordingly adapt motions of the robot as well as implemented production tasks and the resulting collaborative plans.

## References


[1] L. Wang, M. Törngren, and M. Onori, "Current status and advancement of cyber-physical systems in manufacturing," *Journal of Manufacturing Systems*, vol. 37, pp. 517–527, 2015.

[2] B. Dworschak and H. Zaiser, "Competences for cyber-physical systems in manufacturing – first findings and scenarios," *Procedia CIRP*, vol. 25, pp. 345–350, 2014, 8th International Conference on Digital Enterprise Technology - DET 2014 Disruptive Innovation in Manufacturing Engineering towards the 4th Industrial Revolution.

[3] Y. Horiguchi, C. M. Burns, H. Nakanishi, and T. Sawaragi, "Visualization of control structure in human-automation system based on cognitive work analysis," *IFAC Proceedings Volumes*, vol. 46, no. 15, pp. 423–430, 2013, 12th IFAC Symposium on Analysis, Design, and Evaluation of Human-Machine Systems.

[4] P. Fantini, M. Pinzone, and M. Taisch, "Placing the operator at the centre of industry 4.0 design: Modelling and assessing human activities within cyber-physical systems," *Computers & Industrial Engineering*, vol. 139, p. 105058, 2020.

[5] A. Umbrico, A. Orlandini, A. Cesta, S. Koukas, A. Zalonis, N. Fourtakas, D. Andronas, G. Apostolopoulos, and S. Makris, "Towards user-awareness in human-robot collaboration for future cyber-physical systems," in *2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2021, pp. 1–8.

[6] D. Andronas, A. Argyrou, K. Fourtakas, P. Paraskevopoulos, and S. Makris, "Design of human robot collaboration workstations – two automotive case studies," *Procedia Manufacturing*, vol. 52, pp. 283–288, 2020, system-Integrated Intelligence – Intelligent, Flexible and Connected Systems in Products and ProductionProceedings of the 5th International Conference on System-Integrated Intelligence (SysInt 2020), Bremen, Germany.

[7] K. Rajan and A. Saffiotti, "Towards a science of integrated AI and Robotics," *Artificial Intelligence*, vol. 247, pp. 1 – 9, 2017.

[8] S. Makris, *Cooperating Robots for Flexible Manufacturing*, Springer, Ed. Springer International Publishing, 2021, no. 1.

[9] S. Lemaignan, M. Warnier, E. A. Sisbot, A. Clodic, and R. Alami, "Artificial cognition for social human–robot interaction: An implementation," *Artificial Intelligence*, vol. 247, pp. 45–69, 2017.

[10] F. Ingrand and M. Ghallab, "Deliberation for autonomous robots: A survey," *Artificial Intelligence*, vol. 247, pp. 10 – 44, 2017, special Issue on AI and Robotics.

[11] M. Ghallab, D. Nau, and P. Traverso, "The actor's view of automated planning and acting: A position paper," *Artificial Intelligence*, vol. 208, pp. 1 – 17, 2014.

[12] N. Mandischer, T. Huhn, M. Hüsing, and B. Corves, "Efficient and consumer-centered item detection and classification with a multicamera network at high ranges," *Sensors*, vol. 21, no. 14, 2021.

[13] A. Umbrico, A. Orlandini, and A. Cesta, "An ontology for human-robot collaboration," *Procedia CIRP*, vol. 93, pp. 1097–1102, 2020, 53rd CIRP Conference on Manufacturing Systems 2020.

[14] S. Borgo, A. Cesta, A. Orlandini, and A. Umbrico, "Knowledge-based Adaptive Agents for Manufacturing Domains," *Engineering with Computers*, vol. 35, no. 3, pp. 755–779, 2019.

[15] S.-O. Bezrucav and B. Corves, "Modelling automated planning problems for teams of mobile manipulators in a generic industrial scenario," *Applied Sciences*, vol. 12, no. 5, 2022.

[16] A. Umbrico, A. Orlandini, A. Cesta, M. Faroni, M. Beschi, N. Pedrocchi, A. Scala, P. Tavormina, S. Koukas, A. Zalonis, N. Fourtakas, P. S. Kotsaris, D. Andronas, and S. Makris, "Design of advanced human-robot collaborative cells for personalized human-robot collaborations," *Applied Sciences*, vol. 12, no. 14, 2022.

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

[18] S. Pellegrinelli, A. Orlandini, N. Pedrocchi, A. Umbrico, and T. Tolio, "Motion planning and scheduling for human and industrial-robot collaboration," *CIRP Annals - Manufacturing Technology*, vol. 66, pp. 1–4, 2017.

[19] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct 2001.

[20] *Improved AI Planning for Cooperating Teams of Humans and Robots*. Zenodo, 2020.

[21] N. Guarino, *Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy*. IOS press, 1998, vol. 46.

[22] R. E. Fikes and N. J. Nilsson, "Strips: A new approach to the application of theorem proving to problem solving," *Artificial intelligence*, vol. 2, no. 3-4, pp. 189–208, 1971.

[23] D. Mcdermott, M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld, and D. Wilkins, "PDDL - The Planning Domain Definition Language," CVC TR-98-003/DCS TR-1165, Yale Center for Computational Vision and Control, Tech. Rep., 1998.

[24] M. Cialdea Mayer, A. Orlandini, and A. Umbrico, "Planning and execution with flexible timelines: a formal account," *Acta Informatica*, vol. 53, no. 6-8, pp. 649–680, 2016.

[25] P. H. Morris, N. Muscettola, and T. Vidal, "Dynamic Control of Plans With Temporal Uncertainty," in *International Joint Conference on Artificial Intelligence (IJCAI)*, 2001, pp. 494–502.

[26] D. Ballard, "Generalizing the hough transform to detect arbitrary shapes," *Pattern Recognition*, vol. 13, no. 2, pp. 111–122, 1981.

[27] E. Matheson, R. Minto, E. G. G. Zampieri, M. Faccio, and G. Rosati, "Human–robot collaboration in manufacturing applications: A review," *Robotics*, vol. 8, no. 4, 2019.

[28] E. Helms, R. D. Schraft, and M. Hagele, "rob@work: Robot assistant in industrial environments," in *Proceedings. 11th IEEE International Workshop on Robot and Human Interactive Communication*, 2002, pp. 399–404.

[29] S. Borgo and C. Masolo, "Foundational choices in DOLCE," in *Handbook on Ontologies*, ser. International Handbooks on Information Systems, S. Staab and R. Studer, Eds. Springer, 2009.

[30] A. Umbrico, A. Cesta, M. Cialdea Mayer, and A. Orlandini, "PLATINUm: A New Framework for Planning and Acting," *Lecture Notes in Computer Science*, pp. 498–512, 2017.

[31] G. Antoniou and F. van Harmelen, *Web Ontology Language: OWL*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 67–92.