Journal of Machine Engineering, 2024, Vol. 24, No. 2, 44-55
ISSN 1895-7595 (Print)   ISSN 2391-8071 (Online)

Received*: 31 Marz 2024* / Accepted: *21 May 2024* / Published online: *27 May 2024*

AI functionality,
collaborative robot,
machine vision,
adaptive grippers

Madis MOOR[1,2*],
Martins SARKANS[1],
Tavo KANGRU[2],
Tauno OTTO[1],
Juri RIIVES[1.]

AI FUNCTIONALITIES IN COBOT-BASED MANUFACTURING FOR PERFORMANCE IMPROVEMENT IN QUALITY CONTROL APPLICATION

Modern manufacturing faces vastly changing challenges. The current economic situation and technological developments in terms of Industry 4.0 (I4.0) and Industry 5.0 (I5.0) force enterprises to integrate new technologies for more efficient and higher-quality products. Artificial intelligence (AI) and Machine Learning (ML) are the technologies that make machines capable of making human-like decisions. In the long run, AI and ML can add a layer (functionality) to make IoT devices more interactive and user-friendly. These technologies are driven by data and ML uses different types of data for making decisions. Our research focuses on testing a cobot-based quality control (CBQC) system that uses smart fixture and machine vision (MV) to determine the cables inside products with similar designs, but different functionality. The products are IoT modules for small electric vehicles used for interface, connectivity, and GPS monitoring. Previous research describes the methodology of reconfiguration of existing cobot cells for quality control purposes. In this paper, we discuss the testing of the CBQC system, together with creating a pattern database, training the ML model, and adding a predictive model to avoid defects in product cable sequence. Preliminary testing is carried out in the laboratory environment which leads to production testing in SME manufacturing. Results, developments, and future work will be presented at the end of the paper.

1. INTRODUCTION

Robot solutions are widely used by small and medium-sized enterprises (SMEs) in every part of the world. SMEs aim to increase the quality of products and throughput, but moreover, reduce the lack of human labour. According to Eurofound, 39% of European manufacturing companies stated limitations in production due to labour shortages. According to the prediction, made by the United Nations analysis, we will witness a decrease of 95 million workers from 2015 to 2050 [1]. The decreasing trend and demographic effect will accelerate

---

[1] Department of Mechanical and Industrial Engineering, Tallinn University of Technology, Estonia
[2] Mechanical Engineering, TTK University of Applied Sciences, Estonia
[*] E-mail: madis@tktk.ee
https://doi.org/10.36897/jme/189169

and force even more rapid usage of smart robotic systems. While industrial robots (IRs) are widely used in high-volume production and large enterprises, SMEs require high flexibility and multi-purpose application robot cells (RCs). For SMEs, a collaborative robot (cobot) is cost efficient and flexible solution which can help to solve the stated problems above. European manufacturing is represented by 98% of SMEs which mainly produce high-mix low–volume products [2], where the speed of retune and flexibility between different positions in production is essential. It is reasonable to implement and adapt parts of I4.0 [3] branches such as Internet of Things (IoT), Adaptive grippers [5], System Simulation, and Integrated Manufacturing [4]. A previous study was conducted by the authors of this article to develop a methodology to reconfigure an existing RC into a quality control system for SME production [6]. The objective of this research is to develop AI features for an existing RC to improve the performance of cobot-based quality control system.

To establish AI features in a RC, an interconnectivity level must be developed between different equipment inside the RC. The system consists of a cobot and its controller, adaptive gripper, smart fixture, and machine vision (MV) system. The information gathered from MV serves great importance in terms of digitalization and takes robot cells to the next level. A product-based database is essential to be developed for the decision-making module. This decision-making module consists of rules and constraints to detect and decide on the ongoing processes inside the robot cell. This study discusses an approach to enable AI features using variable programming methods, allowing quickly adapt changing products in SME production.

The rest of this paper is organized as follows. The second chapter presents the previous research done on similar topics. The third chapter describes the CBQC system communication. The fourth chapter presents the AI features of CBQC programming. The fifth chapter gives an overview of CBQC setup and testing. The final chapter concentrates on discussion and conclusions, followed by future work and acknowledgment.

2. LITERATURE REVIEW

2.1. COLLABORATIVE ROBOTS IN PRODUCTION

The use of cobots in the industry is constantly growing, growing by 31% in 2022 compared to the previous year, reaching a 10% global market share [7]. The main reason is their minimal installation cost into an existing non-robotised production process to perform repetitive tasks and a short user training period [8]. The RC can be characterized by the level of cooperation, where the low and high are as follows. Starting at the lowest, similar setup to a classic industrial robot cell layout, the robot is separated from the person and performs a given task. Cooperation at the highest level, the robot monitors human worker movements and contributes as needed. The implementation costs are corresponding; in the first case, it is possible to use existing “off–the–shelf” solutions, ensuring the minimum costs. On the other hand, in the case of a high level of cooperation, a specific approach to the production process is needed [9].

2.2. ADAPTIVE GRIPPER SYSTEMS

One of the important steps in implementing a collaborative robot in the SME production process is the selection of a tool or end effector. The benefit of the cobot is the ease of ability to be relocated as needed and used to perform various tasks. Depending on the production task change, the shape of the parts or tools that are likely to be gripped also changes, and it is no longer possible to use only one special jig. This, in turn, usually places more complex requirements on the end effector. Widespread grippers use 2 to 6 fingers to be controlled either with an electric, pneumatic, or hydraulic drive, as L. Birglen et al. presented in their review on IR grippers [10]. Kinematically, either parallel or radial gripping is widely used. Reconfigurable grippers have come into use, with the number of fingers, grip, range, geometry, and force being easily changed [5]. For the adaptive gripper systems, finger speed, acceleration, slippage and force/torque sensing and controlling are essential functions. B. Zhang et al. discussed different methods and strategies for grasp process control and condition monitoring in agricultural robots, where force/torque sensing is essential [11].

2.3. IOT MODULE AND TECHNOLOGY

With the widening application of AI functions in SME processes (decision-making, automation, determining and optimizing parameters, and operations), to reduce production costs and lead times, the need for information (capture, storage, retrieval, processing, and redirection to use) from different enterprise levels has increased considerably. IoT technology covers almost all those needs in a small, low-cost, modular form. IoT architecture comprises four main layers [12]: sensing or actuating, network, data processing, and application. All the layers can be built into the module, or when the module size and cost are important, only the first two layers are minimal. The International Data Corporation (IDC) predicts an annual growth rate of 11% for IoT devices, reaching $345 billion by 2027 [13].

2.3.1. IOT DEVICES FOR PRODUCTION MONITORING

Production monitoring is used to verify the process status and movement of units, materials, and products. Data from production cells or warehouses are gathered, prepared, and transmitted for future processing. Data can be collected from motion, position, environmental, or similar sensors. From the manufacturing point of view, Overall Equipment Effectiveness (OEE) or Overall Labor Effectiveness (OLE) are the most common Key Performance Indicators (KPI). To calculate those KPIs, states, speeds and units are measured and reported to the Manufacturing Execution System (MES) [14]. From MES, the data is converted to information, and further decisions or processes can be made.

2.3.2. IMPLEMENTATION IN THE LAST 5-10 YEARS

Online monitoring of production and the implementation of IoT devices are part of the bigger manufacturing digitalization process. Digitalization has grown significantly in the last decade, reaching a point where there is a desire and will to innovate, but its implementation is significantly affected by the market's uncertainty. 31% of companies that have gone through the digitization process admit that they still collect data through a non-digital process. In addition, entrepreneurs are aware of the possibilities of MES, but 37% lack resources and 34% lack the budget to implement MES [15]. The current situation can also be seen as a momentary slowdown of rapid growth in preparation for the next sprint. Virtual reality, simulation, augmented reality, cyber-physical systems (CPS), artificial intelligence (AI), Internet of Things (IoT), Industrial Internet of Things (IIoT), cloud computing and big data are listed as key technologies of the future industry [16]. The research mentioned above supports the background for our study and using its elements, we have adapted AI functionalities for improving the performance in quality control applications.

3. CBQC SYSTEM COMMUNICATION

3.1. CBQC SYSTEM LAYOUT AND MODEL

For creating a better understanding of a CBQC system, a virtual model of the system is presented, see Fig. 1. We can separate four main areas of the system: input, process, output, and defective area. The process area consists of system equipment and is the most important part of the system. It consists of a cobot Omron TM5-900, adaptive gripper Robotiq 2F-140, MV Cognex In-Sight 7905C, and a smart fixture.

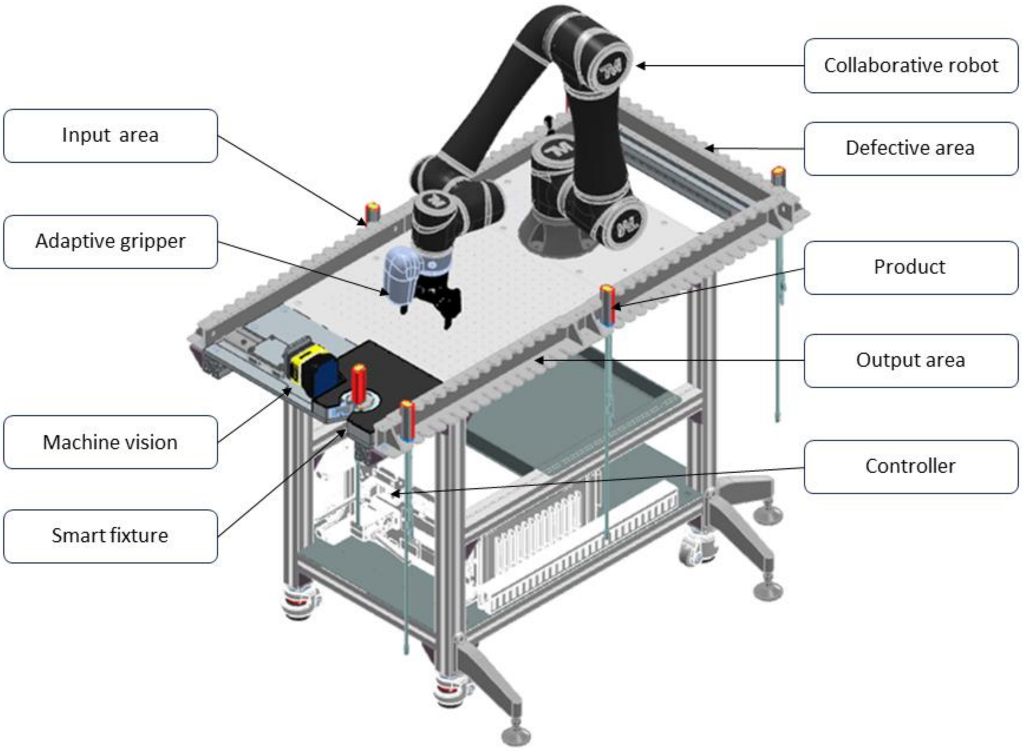


Fig 1. CBQC system areas

3.2. SYSTEM LIMITATIONS

The system developed is built on existing hardware and software available in the university and therefore has some limitations compared to automation system integrators. Additionally, for the hardware mentioned above, the software used for simulation and programming is the following: Omron TMFlow for cobot programming, RoboDK for initial simulation and layout design, Cognex In-Sight Explorer software for MV camera programming, ModPoll software for Modbus communication testing and evaluation.

For the communication between system components physical I/O (Input/Output) connections were used and Modbus communication was established between the MV camera and robot controller. Physical (I/O) connections between the system devices for communication (number of inputs and number of outputs and their functions) are shown in Table 1. For the robot controller, there are available 16 physical inputs and 16 physical outputs for communication with external devices. Also, different communication protocols (Modbus TCP/RTU, Ethernet TCP/IP, RS232, Profinet) are available.

Table 1. I/O values for cobot controller

Input nr	Function		Output nr	Function
DI 0	Sensor for rotational cylinder position 1 detection		DO 0	clockwise rotation of the product in front of the camera (position 1)
DI 1	Sensor for rotational cylinder position 2 detection		DO 1	Counter clockwise rotation of the product in front of the camera (position 2)
DI 2	reserved		DO 2	Selection of MV camera job (bit 1)
DI 3	reserved		DO 3	Selection of MV camera job (bit 2)
DI 4	reserved		DO 4	MV camera trigger. Gives a signal to the camera to take a picture.
DI 5	Reading the camera job result. If the job was OK, the result is 1, if NOK, the result is 0		DO 5	switching on compressed air for small cylinders for product gripping
DI 6	reserved		DO 6	reserved
DI 7	reserved		DO 7	reserved

3.3. SYSTEM INTEGRATION

The cobot, cobot controller, MV system and fixture are connected and integrated by considering mechanical constraints and by establishing communication between system components. Physical outputs are needed for controlling the solenoids for pneumatical cylinders (fixing the product, turning the product in front of the camera) and for sending signals (triggering, program number) to the MV camera. Physical inputs are for detecting the position of the product (cylinder positions 1 and 2) and for reading the signals (camera job result) from the MV camera. Modbus communication was used for reading cobot current position coordinates to calculate new program positions and for communication with the MV camera.

As system integration is a very complex task, the important steps are shown below to be taken to carry out a similar integration project efficiently. The steps are as follows:

1. selection and design of necessary system components (in our case SolidEdge);
2. simulation of the system by using a simulation environment (in our case RoboDK);
3. assembly and connection of system components mechanically, testing cobot reachability;
4. creation of the MV camera program by using suitable software (in our case Cognex In-Sight);
5. creation of the robot program taking into account modularity and parametric programming (in our case Omron TMFlow);
6. establishing physical connections (I/O) and communication between system components, testing the communication (in our case ModPoll, WireShark);
7. testing of the programs and communication to ensure correct functioning.

4. AI FEATURES OF A CBQC PROGRAMMING

4.1. COLLABORATIVE ROBOT PROGRAM STRUCTURE

In this study, the cobot program follows a certain workflow [17] and has been divided into different levels of programs: main, sub, and decision program, see Fig. 2. The main program serves as a master, having full control over subprograms. The structure follows a sequential routine with a set of rules and limitations. There is a limitation in the input area, where the number of products may vary from 15 to 45 products. Another rule is the check-up for adaptive gripper parametric status. Additionally, the force sensing available for the cobot can also be used to recognize collisions at the end-effector level [18]. Sub “2_0_Home” carries out force control and open range for the gripper but is only executed once in a cycle. This is achieved by decision module 2, located between tested products (ON/NOK) and the next loop of the cycle.

The motion movements are classified according to the location of the CBQC system layout to simplify and reduce the time consumption of necessary adjustments in motion parameters or the process itself. Preliminary programming may be more time-consuming than conceptional programming, but on the other hand, the concept allows to detection and repair of any issues which may influence the successful process execution. The schematic presentation of a program structure helps to understand the complexity of a program and focus on essential activities. In this study, the essential focusing points are on sub-programs: “Pick_up”, “CAM_trigger”, “Tested_OK”, and “Tested_NOK”, see Fig. 2.

In reality, the program layout is more linear, see Fig. 3, and the previous schematic structure is the base of the master program activities. Based on the relationships among robots, human operators, and processes, collaborative tasks can be classified into four categories [19]: independent, sequential, simultaneous, and supportive. Our study focuses on sequential collaboration where every program in the master program list is equipped with indirect functions such as „display” and „voice”. These functions create a far better understanding between the operator and the CBQC system. The display function enables an option for monitoring the process on site (status of local and global variables) and offers the possibility to monitor these features online, using cloud technology. The Voice function offers an informative overview to any operator near the CBQC system. Since the system is designed to production flow and stand between manual workplaces (EOL testing and glueing), productivity is affected by cooperation. Both productivity and ergonomic performance are of significant importance in manufacturing [20].

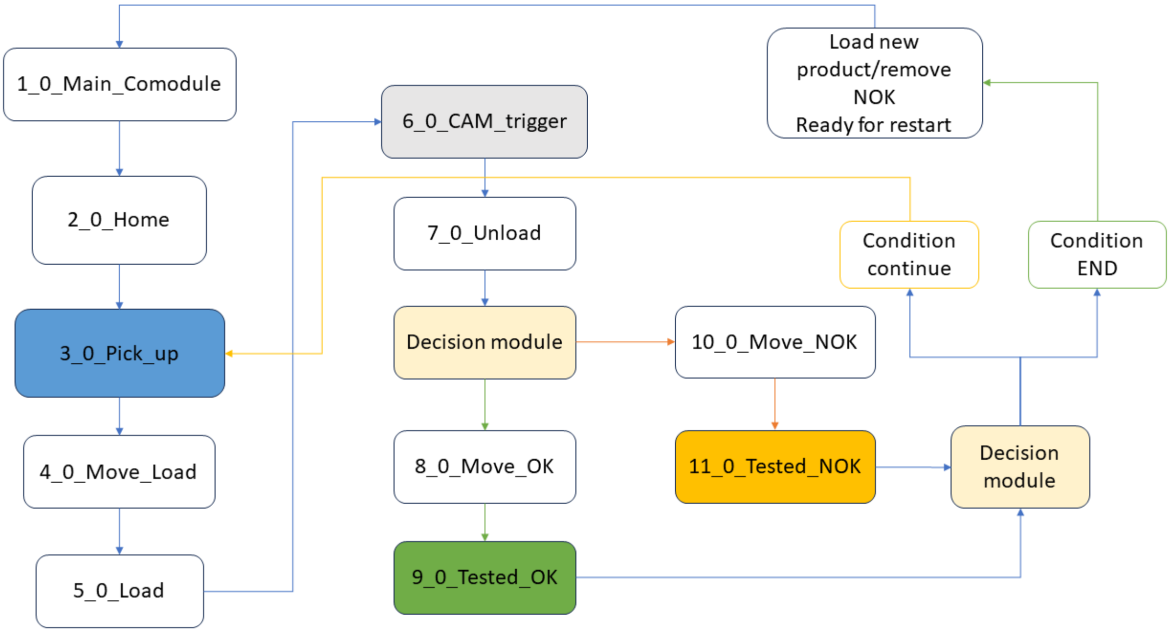


Fig 2 A schematic view of program structure

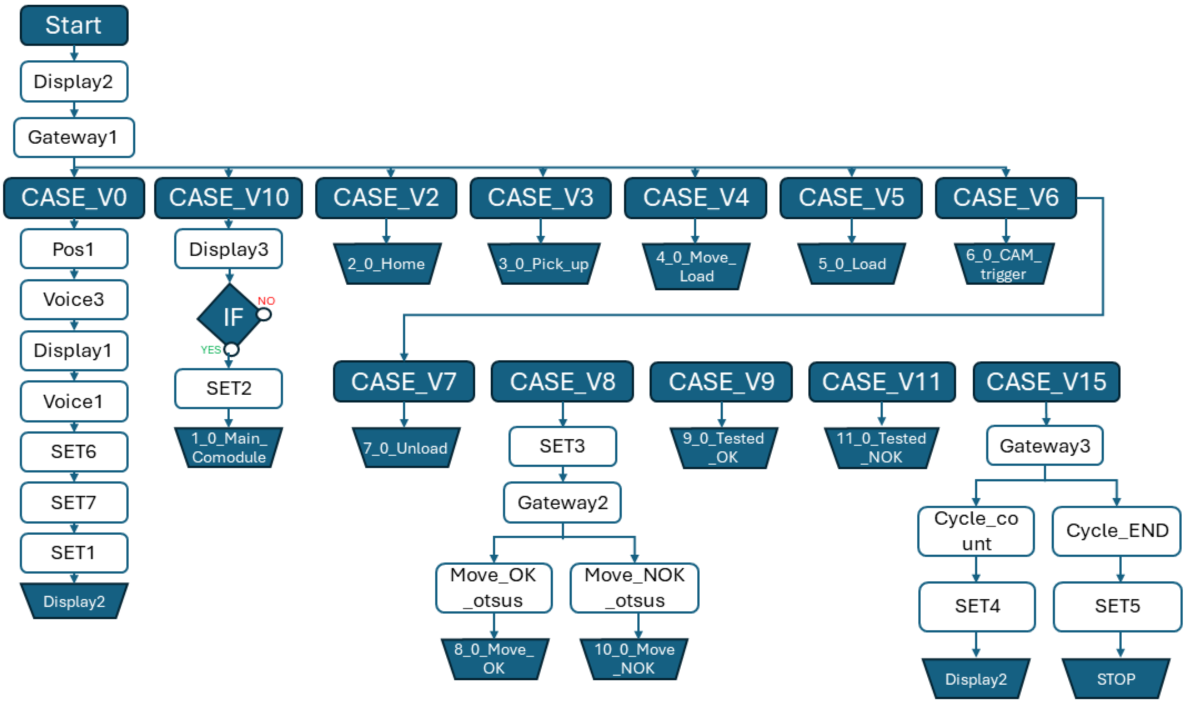


Fig. 3. An example of master program – Main_Comodule

4.2. LOGIC AND VARIABLES

A cobot program with different levels does not make the program shorter. It simplifies the program and ensures easy following. The logic of our research cobot program uses a different type of variable method: local variables, global variables, and Modbus protocol. Modbus is a request-response protocol implemented using a master-slave relationship. In a master-slave relationship, communication always occurs in pairs—one device must initiate a request and then wait for a response the initiating device (the master) is responsible for initiating every interaction [21]. Local variables function as though they were temporary attributes defined in a specific logic section which disappear when the logic section is finished executing. Local variables are useful for testing variables in loops and storing locally used, unique values for each entity at the current location [22]. A global variable, on the other hand, is declared outside any functions or tasks, and therefore typically appears at the very top of a program. Because they are declared at a level broader than any task or function, all functions and tasks can "see" global variables, and they do not lose their value even after a function or task ends [23].

In our study, we use Modbus protocol to read the present values of the cobot linear axis and the information is used for next motion calculations. For example, this logic is used in most local variables in combination with Modbus read and global variables. Modbus communication explanation is shown in Table 2.

Global variables have control over essential sub-programs stated above, see 4.1. In terms of the number of program motion positions, global variables allow for a decrease of motion steps over five times. We have used 66 variables all together throughout the 11 programs and this example is stated according to 15 products handled in the CBQC system. It is possible to increase the number of products to 45 without making any design changes to the system layout. An example of global variable functionality is shown in Table 3.

Table 2. Modbus communication

Definition	Address	Type	Explanation
Preset_tcp_y	7051	Float	Read current position value of Y axis
Preset_tcp_x	7037	Float	Read current position value of X axis
Preset_tcp_z	7041	Float	Read current position value of Z axis

Table 3. Example of global variables in CBQC system

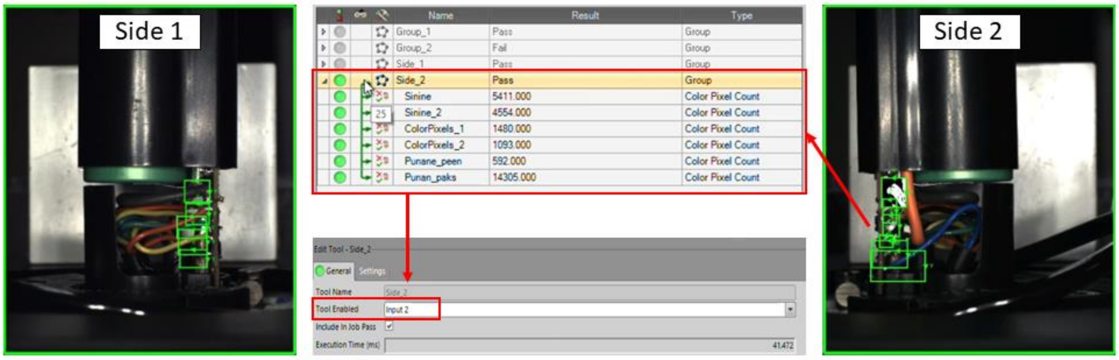
Module	Name	Type	Value	Explanation
1_0_Main_Comodule	G_num3	Int	0	Main program - Case decision value – program structure logic
3_0_Pick_up	G_num1	Int	9	Input grip area – counts input product program value
	G_shiftxin1	Float	47	X axis motion IN – Input grip
	G_shiftxout2	Float	50	X axis motion OUT – Input grip
	G_shiftzup3	Float	15	Z axis motion UP (height) – Input grip

9_0_Tested_OK	G_num5	Int	2	Output place area - counts OK product program value
	G_shiftxin5	Float	50	X axis motion IN – Output OK place
	G_shiftzup5	Float	10	X axis motion OUT – Output OK place
	G_shiftxout5	Float	70	Z axis motion UP (height) – Output OK place
11_0_Tested NOK	G_num8	Int	0	Output place defective area – read NOK product value
	G_shiftxin8	Float	52	X axis motion IN – Output NOK place
	G_shiftzup8	Float	15	X axis motion OUT - Output NOK place
	G_shiftxout8	Float	50	Z axis motion UP (height) – Output NOK place
6_0_CAM_trigger	G_num10	Int	0	Camera trigger OK – OK product counter (decision module)
	G_num11	Int	0	Camera trigger NOK – NOK defective product counter (decision module)

4.3. MACHINE VISION PROGRAM STRUCTURE AND DETECTION

MV system triggering is executed using the cobot controller's control box, through digital outputs (DO). This evokes a program from Cognex In-Sight software. The product needs to be visually checked from two positions: 0 degrees and 180 degrees and is achieved using a smart fixture [6], the program follows the same structure. An example used for this study has 6 cables on one side and 6 cables on the other side with different colours. It is also possible that the same colour marker is used on the same side, but different functionality of the product.

A colour database was created in the MV system for each colour. This means that each cable went through colour training in different light conditions and constraint setups. Constraints setup includes a lower and upper boundary variation in the number of pixels being counted, see Fig. 4. As the quality of product functions is essential, correct product cable results must be all true. If any colour is not detected or is out of range, then the product will move to the defective area. While the MV triggering was executed through DO, the feedback for the cobot controller was sent through digital input (DI) channels. True value creates 1 and false value creates 0, which is used in the decision module. The decision module compares the values, and the controller directs to the corresponding sub-program for execution.



Name	Result	Type
Group_1	Pass	Group
Group_2	Fail	Group
Side_1	Pass	Group
Side_2	Pass	Group
Sinine	5411.000	Color Pixel Count
Sinine_2	4554.000	Color Pixel Count
ColorPixels_1	1480.000	Color Pixel Count
ColorPixels_2	1093.000	Color Pixel Count
Punane_peen	592.000	Color Pixel Count
Punan_paks	14305.000	Color Pixel Count

Edit Tool - Side_2

Tool Name: Side_2
Tool Enabled: Input 2
Include In Job Pass: ☑
Execution Time (ms): 41.472

Fig. 2. An example of MV program structure

5. CBQC SETUP AND TESTING

This paragraph presents the overview of the CBQC setup and testing, introducing results. Our research is based on previous results of the authors' article [6], and lots of effort has been directed to programming and logic to meet the requirements of AI functions and increase the system's sustainability.

The proof of concept of the CBQC system has been achieved: Cobot can handle the products, adaptive gripper jaw design offers the required grip without damaging the product, MV detects necessary objects (cables) and colours, and the decision is made and cobot controller executes the information. We have run test cycles with the products with a speed of 40% which is 280 mm/s. The motion speed is the major parameter in this system cycle time and the time consumption for one product is 90 seconds. The motion speed can be increased (previously suggested 250-500 mm/s [6]), keeping sustainability in mind.

Our testing used random placement of different IoT modules from three models. We can state that zero defective or unsuitable models, from 350 products tested, did not pass the CBQC system. On the other hand, a new problem occurred during testing. Due to the changes in the lighting environment during daylight hours, approximately 8% of OK products were marked with NOK and sent to defective area. This problem must be solved with more consistent lighting conditions and decrease the influence of daylight.

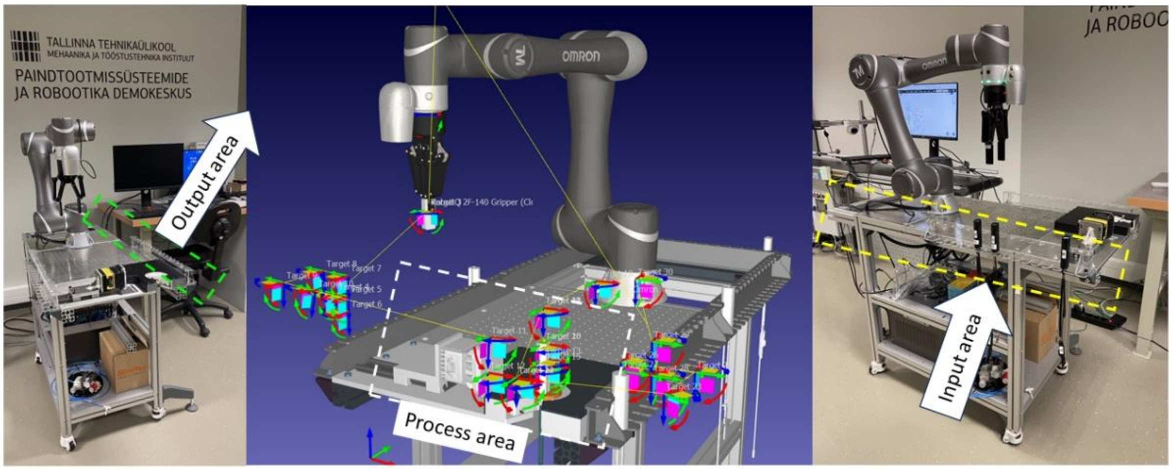


Fig. 3. Virtual and real cobot based quality control system [6]

6. CONCLUSION

In this study, the importance of AI functionalities in cobot-based quality control applications has been discussed. The integration and implementation of different equipment into a functional CBQC system is a complex assignment. Following structural rules and determining system and equipment limitations help to create boundaries for faster integration. AI functionalities such as Modbus protocol, and local and global variables enable to improvement of programming performance in terms of creating the multi-layer communication between system equipment in terms of hardware and software. This, on the other hand, improves the performance of the entire system.

In the practical use case, we have presented an approach how to classify the CBQC system program into modules, allowing a simplified overview and navigation for the user. This helps to solve any occurrences faster in the process or adapt new products into the CBQC in the 60% less of time consumption. Also, we presented the effective use of variables for making the system program five times more compact and easier to reconfigure. Either scaling up the program in terms of products or changing the system parameters. The time consumption for one product is 90 seconds with a speed of 40% of cobot. In terms of MV detection, it takes 2 seconds to detect both sides of the product together with smart fixture motion, where image recognition takes 200 ms per side. For future development, the possibility of machine learning functionality as part of AI functionality should be studied and discussed. During the future testing, we can collect more data (position of cables, changes in lighting environment, and common faults) and train machine vision system to adaptable detection system with real-time data. Renewable pattern database would keep any changes with the product more easily trackable.

ACKNOWLEDGEMENTS

We would like to thank Comodule Ltd. for providing the input information used in our experiments, and Mr. Tanel Eensoo for his support and participation in the implementation. Furthermore, we would like to gratefully acknowledge the contributions of our colleagues from Taltech, IMECC, and TTK UAS who provided insight and expertise that greatly assisted the research.

REFERENCES

[1] Eurofound, 2021, *Tackling Labour Shortages In EU Member States*, Publications Office of the European Union, Luxembourg.

[2] PIESKA S., KAARELA J., MÄKELÄ J., 2018, *Simulation and Programming Experiences of Collaborative Robots For Small-Scale Manufacturing*, 2nd International Symposium on Small-scale Intelligent Manufacturing Systems (SIMS), Cavan, Ireland, 1-4, https://doi.org/10.1109/SIMS.2018.8355303.

[3] BATTH R.S., NAYYAR A., NAGPAL A., 2018, *Internet of Robotic Things: Driving Intelligent Robotics of Future - Concept, Architecture, Applications and Technologies*, 4th International Conference on Computing Sciences (ICCS), Jalandhar, 151–160, https://doi.org/10.1109/ICCS.2018.00033.

[4] MOOR M., VAHER K., RIIVES J., KANGRU T., OTTO T., *Modern Robot Integrated Manufacturing Cell According to the Needs of Industry 4.0,* Proceedings of the Estonian Academy of Sciences, Tallinn, 202, 407–412, https://doi.org/10.1088/1757-899X/1140/1/012034.

[5] FRIEDMANN M., FLEISCHER J., 2022, *Automated Configuration of Modular Gripper Fingers,* Procedia CIRP, 106, 70–75, https://doi.org/10.1016/j.procir.2022.02.157.

[6] MOOR M., SARKANS M., RIIVES J., OTTO T., VAÑÓ J.M., *Methodology for Reconfigurable Cobot-Based Quality Control System for SME Production,* International Journal of Engineering and Technology, https://doi.org/10.7763/IJET (accepted, to be published).

[7] International Federation of Robotics, 2023.

[8] GAMBAO E., 2023, *Analysis Exploring Risks and Opportunities Linked to the Use of Collaborative Industrial Robots in Europe,* European Parliamentary Research Service.

[9] International Federation of Robotics, *2020, Demystifying Collaborative Industrial Robots,* International Federation of Robotics, Frankfurt.

[10] BIRGLEN L., SCHLICHT T., *2018, A Statistical Review of Industrial Robotic Grippers,* Robotics and Computer-Integrated Manufacturing, 49, 88–97, https://doi.org/10.1016/j.rcim.2017.05.007.

[11] ZHANG B., XIE Y., ZHOU J., WANG K., ZHANG Z., *2022, State-of-the-Art Robotic Grippers, Grasping and Control Strategies, as Well as Their Applications in Agricultural Robots: a Review,* Computers and Electronics in Agriculture, 177, https://doi.org/10.1016/j.compag.2020.105694.

[12] SIKDER A.K., PETRACCA G., AKSU H., ULUAGAC S., 2018, *A Survey on Sensor-based Threats to Internet-of-Things (IoT) Devices and Applications,* https://doi.org/10.48550/arXiv.1802.02041.

[13] TORCHIA M.S.M., *2019, IDC Forecasts Worldwide Spending on the Internet of Things to Reach $745 Billion in 2019,* IDC.

[14] RODRIGUES D., CARVALHO P., et al., 2022, *An IoT Platform for Production Monitoring in the Aerospace Manufacturing Industry,* Journal of Cleaner Production, 368, https://doi.org/10.1016/j.jclepro.2022.133264.

[15] PARSEC AUTOMATION CORP., 2023, *Parsec's 2023 State of Manufacturing Survey: Europe.*

[16] SINGH T., SINGH D., 2023*, Factories of the Future: Technological Advancements in the Manufacturing Industry,* Wiley.

[17] MATERNA Z., KAPINUS M., BERAN V., SMRŽ P., ZEMČÍK P., 2018*, Interactive Spatial Augmented Reality in Collaborative Robot Programming: User Experience Evaluation,* 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Nanjing, China, 80–87, https://doi.org/10.1109/ROMAN.2018.8525662.

[18] BIRGLEN L., 2019, *Design of a Partially-Coupled Self-Adaptive Robotic Finger Optimized for Collaborative Robots,* Auton. Robot., 43, 523–538, https://doi.org/10.1007/s10514-018-9802-x.

[19] EL ZAATARI S., MAREI M., LI W., USMAN Z., 2019, *Cobot Programming for Collaborative Industrial Tasks: an Overview,* Robot. Auton. Syst., 116, 162–180, https://doi.org/10.1016/j.robot.2019.03.003.

[20] ZHANG Y.-J., LIU L., HUANG N., RADWIN R., LI J., 2021, *From Manual Operation to Collaborative Robot Assembly: an Integrated Model of Productivity and Ergonomic Performance,* IEEE Robotics and Automation Letters, 6/2, 895–902, https://doi.org/10.1109/LRA.2021.3052427.

[21] NATIONAL INSTRUMENTS CORP, 2024, *What Is Modbus Protocol and How Does It Work?,* Retrieved March 12, https://www.ni.com/en/shop/seamlessly-connect-to-third-party-devices-and-supervisory-system/the-modbus-protocol-in-depth.html.

[22] PROMODEL CORP, 2012, *Building the Model: Advanced Elements – Variables – Local Variables,* https://www.promodel.com/onlinehelp/ProModel/86/Content/Topics/C-06%20-%20Local%20Variables.htm.

[23] CARNEGIE MELLON ROBOTICS ACADEMY, 2024, *Global Variables,* https://www.robotc.net/teaching_rc_cortex_v2/lesson/media_files/hp_global_variables.pdf.