TEM Journal. Volume 12, Issue 4, pages 1929-1938, ISSN 2217-8309, DOI: 10.18421/TEM124-02, November 2023.

Evaluating the Reliability of a Machine Vision System for Collaborative Robots: An Experimental Study in the Industry 4.0 Environment

Jakub Müller [1], Tomáš Broum [1], Miroslav Malaga [1]

[1] *Department of Industrial Engineering and Management, University of West Bohemia, Univerzitní 2732/8, 301 00 Pilsen, Czech Republic*

Abstract **– This research evaluates the reliability of a machine vision system connected to a collaborative robot. In recent scientific papers, the authors have focused on machine vision itself, machine vision systems and related theory in general, along with machine learning methods, and image processing itself. However, there seems to be a missing link between these topics and the industrial robot's accuracy in basic tasks when utilizing machine vision. The experiments conducted, took place within an Industry 4.0 laboratory, where 3D-printed objects were utilized as test subjects. The collaborative robot, equipped with machine vision, performed tasks such as object removal and stacking. The evaluation focused on the success rate of object assembly and grasping. The paper discusses the integration of machine vision technology, previous research on reliability, and the use of a 2D camera for the collaborative robot. The findings contribute to understanding the potential of machine vision in enhancing efficiency and precision in collaborative robot workspaces.**

Keywords **– Collaborative robots, computer vision system, reliability, industry 4.0, experimental study.**

DOI: 10.18421/TEM124-02
https://doi.org/10.18421/TEM124-02

Corresponding author: Miroslav Malaga,
Department of Industrial Engineering and Management, University of West Bohemia, Pilsen, Czech Republic
Email: malaga@fst.zcu.cz

Received: 25 July 2023.
Revised: 16 October 2023.
Accepted: 02 November 2023.
Published: 27 November 2023.

1. Introduction

The introduction of industrial robots marked a significant milestone in the field of manufacturing, introducing increased levels of precision and efficiency.

Over the years, the industrial robots have become integral to various industrial sectors, from automotive industry to electronics manufacturing [1], [2]. However, the ability of robots to work collaboratively with human operators, without the need for safety cages, has remained a persistent challenge. This problem was successfully addressed with the introduction of collaborative robots, that revolutionized the concept of human-robot interaction by enabling safe and efficient collaboration in shared workspaces [3].

An important element for the successful integration of industrial/collaborative robots is machine vision technology. Machine vision empowers robots with visual perception capabilities, allowing them to interpret and comprehend their surroundings. By using cameras and advanced image processing algorithms, robots equipped with machine vision systems can detect, recognize, and analyze objects, ensuring precise manipulation, inspection, navigation, or other required actions [4], [5], [6].

The objective of the paper is to present an evaluation of the reliability related to computer vision system connected to collaborative robot. The evaluation is based on the experiments performed in the laboratory of Industry 4.0 at the University of West Bohemia in Pilsen. The specific objects needed for the experiments were printed on a 3D printer. The objects serve as building blocks for experimental assembly. The collaborative robot, through the vision process, is able to remove and stack objects over the entire field of view of the camera. The final evaluation is concentrated on the experiments of reliability in terms of the success rate of assembling and grasping each object.

2. Literature Review

This chapter focuses on providing a basic definitions of machine vision and state of art of the important research related to the topic of machine vision reliability.

The definition of machine vision, according to the Automated Imaging Association, encompasses all industrial and non-industrial applications in which the combination of hardware and software provides devices with operational instructions based on image capture and processing [7]. According to the definition by Robotics Tomorrow, machine vision is the ability of a computer to perceive the environment. It utilizes one or more cameras with analog-to-digital conversion and digital signal processing. The image data is transmitted to a computer or a robot's control unit [8]. According to the definitions can be generally said that machine vision is ability of a computer / robot to perceive its environment.

The reliability of machine vision systems has been researched in different ways by different authors. This topic is mentioned in the paper named Reducing Pseudo-error Rate of Industrial Machine Vision Systems with Machine Learning Methods [9]. The paper provides examples of possible applications of machine learning algorithms in manufacturing linked to the machine vision systems, as well as reducing the pseudo-error (false positive) rate of machine vision quality control systems. It shows that even the simplest algorithms and models can be effective in reducing the errors of machine vision systems and specifically in the paper are used convolutional neural networks that reduced the degree of pseudo-error of the presented system [9].

Other authors are concentrating on automated fault detection that has significant importance to the computer vision industry. In their paper named Automated Failure Detection in Computer Vision Systems [10] they used deep neural network to detect computer vision failures in vehicle detection tasks. They train neural network to learn to the estimates the output quality of vehicle detector and evaluates the results of comparison between their vehicle detector and human-annotated data [10].

In the paper named Developing a Machine Vision Inspection System for Electronics Failure Analysis [11] is the perspective of machine vision errors seen in the field of electronics industry. The papers aims to develop a machine vision image recognition system for intelligent decision analysis to resolve the seriousness of solder ball cracking. The verification of results demonstrates that the developed system and failure analysis achieve a consistency rate of over 85% in judging the severity of cracking.

The system exhibits a false alarm ratio of approximately 9% and an escape ratio of around 16% [11].

The next relevant paper names: A Comparative Study of Machine Vision Based Methods for Fault Detection in an Automated Assembly Machine [12]. In the paper both normal and abnormal assembly machine conditions are compared. For comparison are used three methods that are afterwards evaluated [12].

The last relevant paper founded named Predicting Failures of Vision Systems [13]. Overall, the paper emphasizes the significance of addressing failures in computer vision systems. The authors promote specific metrics to evaluate failure prediction. Then the authors propose a straightforward approach called ALERT, which can predict the accuracy or failure of various computer vision systems on individual input images. The proposed ALERT approach shows is evaluated on four datasets [13].

Mentioned related papers [9], [10], [11], [12], [13] are concentrating purely on machine vision, but they do not include and therefore not evaluate connected industrial robot linked to the machine vision system. The reliability of grasping related to the robot is therefore not evaluated. The advantage of the papers [9] and [10] are the number of tested samples, the paper [9] tested approx. 23 000 pcs. The paper [10] tested approx. 8 000 video frames.

Other perspective on the problematics of vision system errors presents Erickson-Davis in the paper What it is to see: Artificial vision as constitutive interaction [14]. The paper concentrates on Artificial vision system of person, specifically visual prosthesis devices - devices that electrically stimulate the visual system to restore vision of people who have lost it. It evaluates errors of current systems and proposes the new approach to overcome them [14].

The paper [14] is also concentrating on the field not so related to the industrial machine vision, but more to the field of human artificial vision.

The mentioned papers show the potential of the research in the topic of testing and evaluation of the reliability related to computer vision system connected to collaborative robot.

3. Using a 2D Camera for a Collaborative Robot

The experiment was conducted using the Industry 4.0 workplace, which is part of the Department of Industrial Engineering and Management, Faculty of Mechanical Engineering, University of West Bohemia in Pilsen. The workplace (Figure 1) consists of a work area (conveyor belt) and the collaborative robot itself. The rest of the equipment shown in the figure is used for another experiment and is therefore not shown below.

For the purpose of the experiment, the workspace was divided into three areas. The workspace consists of an area dedicated to the base, an area for assignment and rewiring objects, and an area for removing objects. A camera and a collaborative gripper are installed on the collaborative robot.

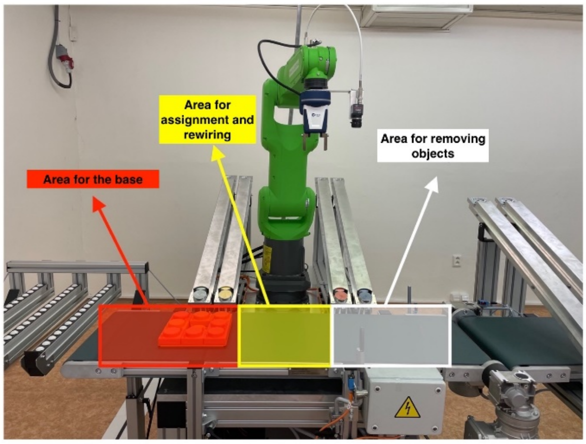


Figure 1. Workplace

The collaborative robot in the workplace is the Fanuc CR-7iA/L. This series of collaborative robots is based on the LR series. The robot has a maximum payload of 7 kilograms, a range of 911 millimeters and works in six axes. It supports the latest Fanuc features (iRVision and Force Sensing). The robot is equipped with a small R-30iB controller [15]. The controller also provides full connectivity via Ethernet, allowing easy connection of robots, remote computers and other hardware [16]. The collaborative gripper is from Schunk (Figure 2). Safety is ensured through current limiting. Schunk's interface works with collaborative robots from Kuka, Fanuc and Universal Robots [17].

Figure 2. Used collaborative gripper by Schunk

The camera system used is iRVision 2D is also from Fanuc, which uses the Fanuc SC130EF2C camera to allow the robots to see.

This camera, positioned at the robot's end effector (Figure 2), is capable of detecting colors. iRVision is simple to use and can be operated without complex programming or expertise. iRVision is fully integrated and does not require external equipment. The system eliminates the need to settle the product in a precise position for the robot to grasp.

3.1. Description of the Experiment

The robot's task will be to assemble a 3D spatial assembly with existing layers. Two types of assemblies were planned, with the working names of the assemblies chosen as garage and house. The assemblies will be composed of 3 types of objects (Figure 3) and a base (Figure 4). The 3x3 base and objects were printed on a 3D printer at the University of West Bohemia.

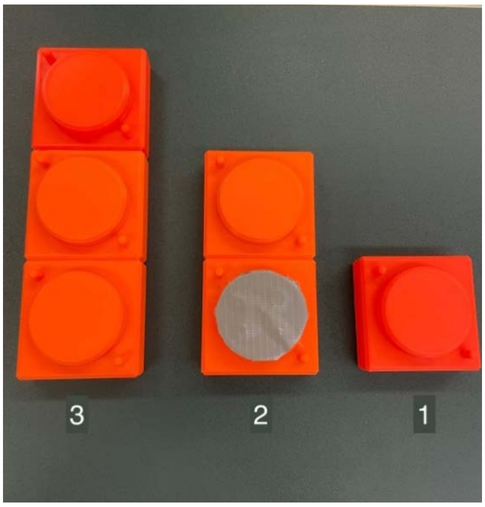


Figure 3. 3D printed assembly objects

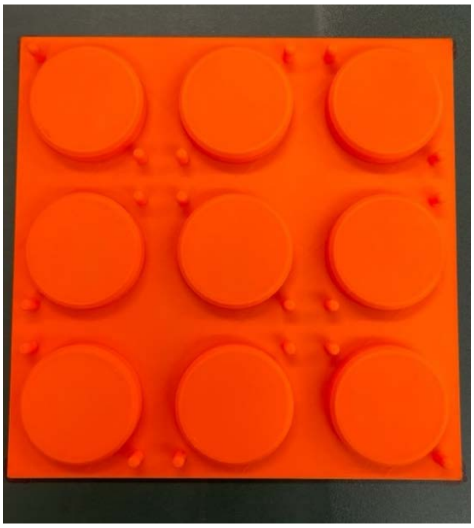


Figure 4. 3D printed base

The actual process of assembling the objects is started by recognizing whether the robot will build a garage or a house.

The recognition is done by a camera, where a piece of paper with the capital letter G for the garage or D for the house is presented on the line being scanned (Figure 5).

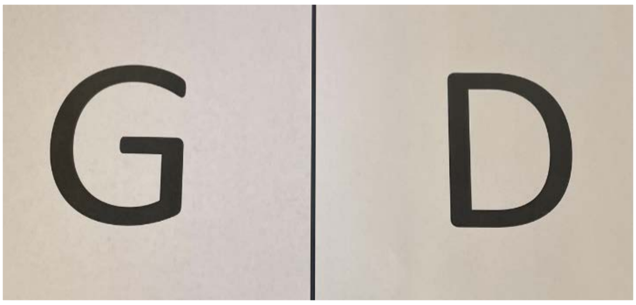


Figure 5. Decision papers for a robot with a camera

The vision process decides which program is executed. The letter G indicates the execution of the Garage program, and the letter D indicates the execution of the House program. In the absence of either letter, the robot will not proceed.

Figures 6 and 7 show the different floors of the house and garage.

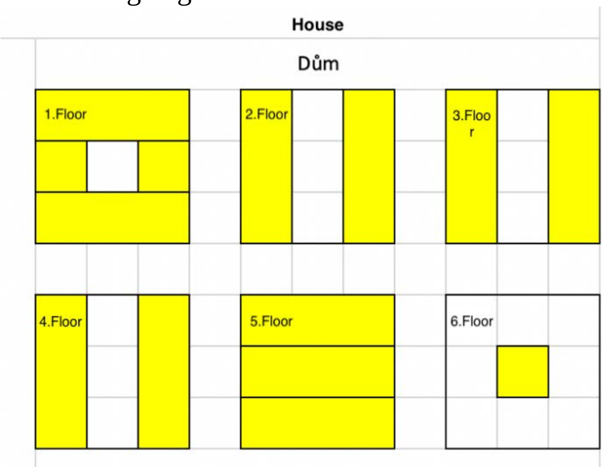


Figure 6. The house plan

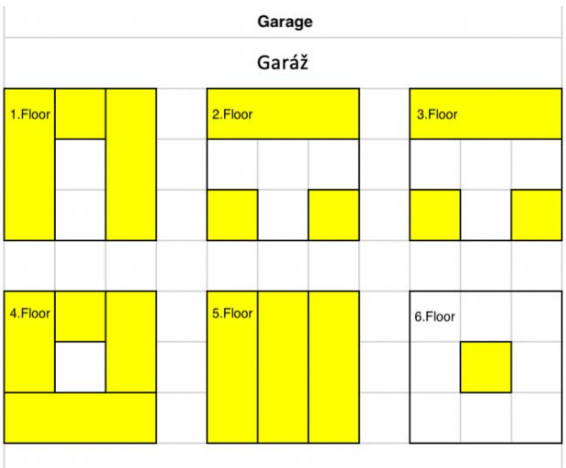


Figure 7. The garage plan

After the input is loaded, the corresponding program is started and folding begins. Folding consists of three different objects.

The objects must be fed manually to the line. In the absence of an object, a base, or a build error, the robot will stop and declare an error. The goal is to completely assemble the house or garage.

The objects in Figure 3 are taken by the robot from the area imaged by the camera. For object 2, there was a problem when there was an overlap and the object started to overbalance (the case for the garage build Figure 7). This problem was solved by drilling holes on one side and adding weights to the object. For Object 2, we have to solve the removal in two cases so that the robot always grabs the correct side of the object, while not overturning and cutting the cables (cables to connect the gripper and the camera).

3.2. Camera System Setup

In total, 7 processes had to be set up for scanning (letter G, letter D, base, object 1, object 2 twice and object 3). In Figure 8 a total of 4 frames are shown.

- The blue frame shows what type of process setup we chose " 2-D Single-View Vison Process" and then locators are added to better recognize the object(s). The "Snap Tool" is used to set up the camera and the "GPM Locator Tool" is used to learn the object.
- In the yellow box is the area we are scanning.
- In the orange frame we have the detailed settings of the "GPM Locator Tool". The GPM Locator Tool settings have a lot of values/pointers and we are trying to get the best object recognition. In the locator, we teach a process to recognize the object with the goal of the highest score.
- In the green box you can see the object's rating, which includes the score, which has a maximum value of 100.

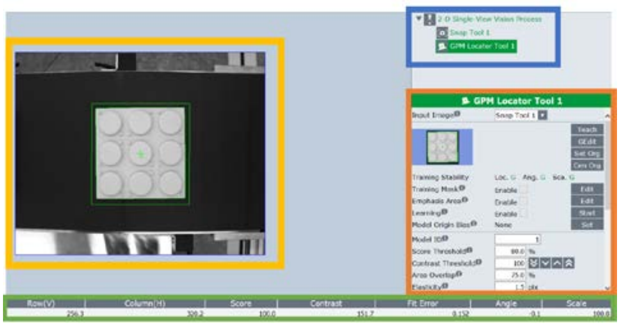


Figure 8. Score of the base

3.3. Experimental Procedure

The purpose of the experiment is to create a working system and then evaluate the build/grab quality. To create a functional system, it is necessary to determine the layout of programs, subroutines, and vision processes so that in the end the whole system can work by running one program.

The experiment should recognize the assignment and then assemble the assemblies called “house” or “garage” from the objects. The resulting assemblies have a predefined structure.

Fanuc's iRVison environment will be used to create the vision processes. The focus will be on creating a first high quality process that will then be used for the rest of objects which will be "learned" from it. The scores for the processes must not be lower than 95% (100% = best) in the iRVison environment. The scores represent the quality of the recognition of each object.

For the programming part, a hierarchy of programs/subroutines was created to keep programming simple and not create too many unnecessary programs. The hierarchy must be followed when creating programs, see Figure 9. The creation of programs should go from Program DP1, DP2, ... to top of the hierarchy, as shown in Figure 9). Description of the programs is following:

- DP1, DP2, DP3 are programs related to grasp of objects 1, 2, 3.
- DP_Base is program related to identification of the base platform.
- DP_Main is the main program including all the subprograms.
- DP_Dum is the program related to build the “house” assembly.
- DP_Garaz is the program related to build the “garage” assembly.
- DP1_Re-grasp, DP2_Re-grasp, DP2_Re-grasp are programs related to regrasping described in the next paragraph.

The re-grip programs are added as they will improve the quality of the grip for the build and will be finished separately so that in case of discarding, only the program calls from DP1, DP2, ... can be removed and the whole programs do not have to be rewritten. The re-grip programs will be created after the completion of DP1, DP2, ... programs and then continue the hierarchy from the bottom. The re-grasping programs should be an intermediate step between the first grab and settling on the assembly. The grasped object will be placed on the area between the assembly and the objects to be taken, and then the camera will find a position where the object is in the center of the field of view and perform a new take, which is more accurate and reduces the possibility of an error in the fit to the assembly.

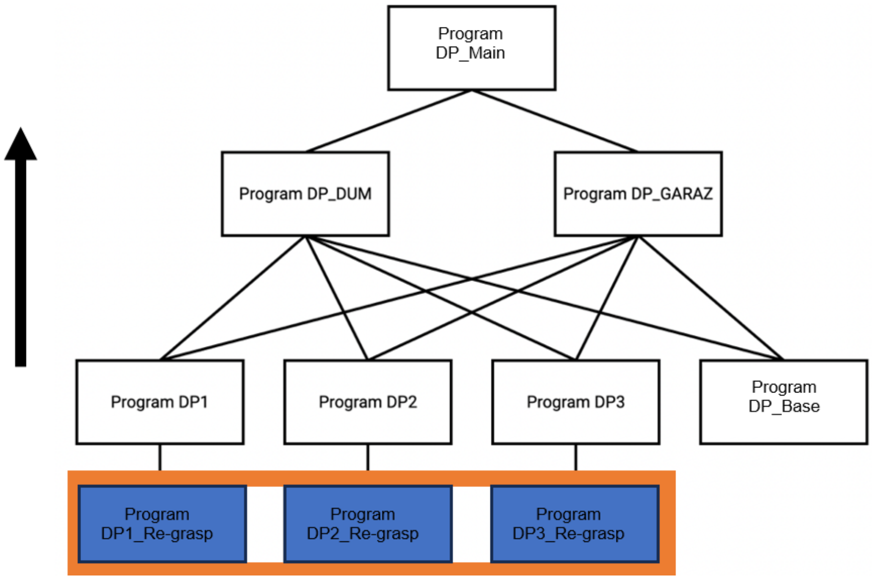


Figure 9. Hierarchy of programs/subroutines

Flowcharts (7 in total) were also created to ensure that the programs met all the requirements and alert the user in case of an error. The flowchart shows the logic of the program. It is used to graphically represent the steps of the algorithm and contains patterns, where each pattern has its own specification.

The flowchart is for the program "DP_Main" (Figure 10). The description of the program "DP_Main" is as following.

In the beginning, we load the vision processes "DPDUM" and "DPGARAZ".

These processes have a recognition assignment stored in them (letters D and G). In the next step, we assign "DPDUM" to "vision get_offset" i.e., the program will compare the process "DPDUM" with the camera image in the next step. If the process "get_offset" and the camera match, the program "DP_DUM" is called, if not, we assign the process "DPGARAZ" to "get_offset". If the process and the detected camera image match, the program "DP_GARAZ" is called, if not, we use the jump command "1" and move to the beginning again, where the program should list what the error was.

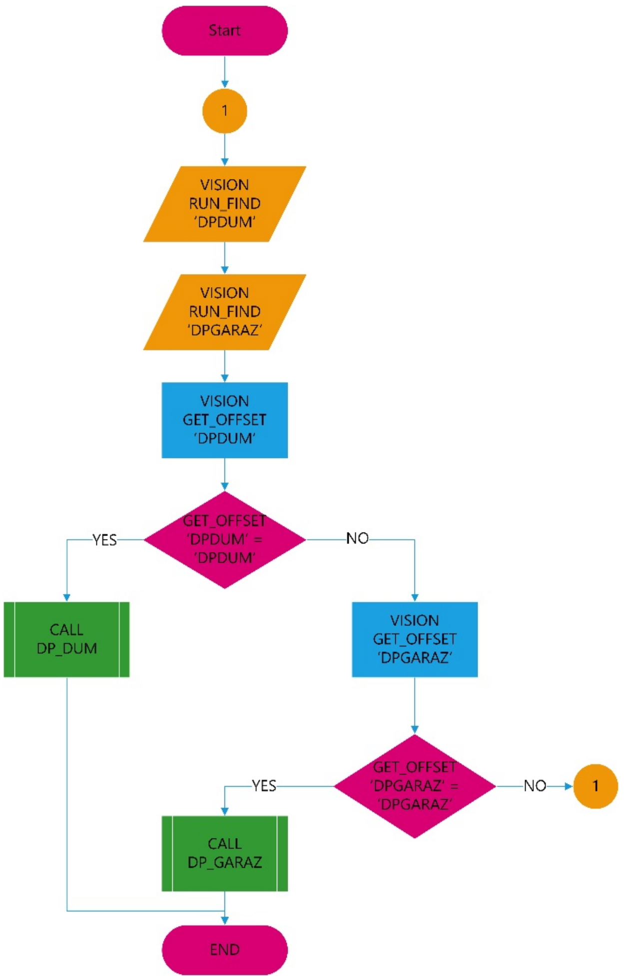


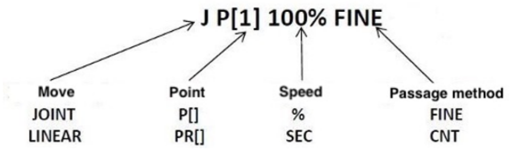
Figure 10. DP_Main program flowchart

3.4. Programming Collaborative Robot

Fanuc has its own programming language called KAREL. KAREL is a lower-level programming language similar to Pascal [18]. There are two terms used in robot programming [18].

- The first variant is online programming. Online programming means that a human is working with a real robot, there is direct interaction. This variant was also used in this practical part. The robot directly drives to the desired positions, either with the help of a control pendant or the human can guide the robot to the position (in this case, but we still have to use the pendant to store the position)
- The second option is offline programming. Offline programming takes place in an application that virtually replaces the workstation. This type of programming is mainly used in companies where they cannot afford to stop the line for testing or learning the robot.

The following Figure 11 shows the general basic movement in the Fanuc Karel programming language with additional information.



Method of movement:
- JOINT - the robot reaches the point by the shortest and therefore the fastest way
- LINEAR - linear movement along the axes

Point:
- P[] - point stored only in the program
- PR[] - position register stored in memory (e.g. a point for taking pictures)

Passage method:
- FINE - the robot arrives exactly at the point and stops at it
- CNT - CNT has values 0 - 100 and is determined by how close the robot gets to the point, 0 = FINE, the robot does not stop at the point

Figure 11. General basic movement in the Fanuc Karel programming language

3.5. Sample of the Main Program

An example of the generated code of the program DP_Main in the Fanuc Karel programming language is shown in Figure 12.

```
LBL[2]
WAIT (DI[16:OFF:TLACITKO OBSLUHA])
J  P[1] 100% FINE
VISION RUN_FIND 'DPDUM'
VISION RUN_FIND 'DPGARAZ'
VISION GET_OFFSET 'DPDUM' VR[3] JMP LBL[1]
CALL DP_DUM
JMP LBL[99]
LBL[1]
VISION GET_OFFSET 'DPGARAZ' VR[3] JMP LBL[98]
CALL DP_GARAZ
JMP LBL[99]

LBL[98]
UALM[5]
JMP LBL[2]

LBL[99]
[End]
```

Figure 12. DP_Main program in Fanuc KAREL programming language

The explanation of the code sample is as follows. At the beginning of the code is "LBL2" because of skipping in case the robot does not load the input. Then it waits for the main line button, which starts the whole system. The robot then goes to the point. The point was chosen so that the quality of the photo could be as good as possible. The main program works with the vision process, where it distributes what construction is done. The condition of which program to execute was solved using "JMP LBL". If "GET_OFFSET" does not load the corresponding vision process "DPDUM", the program jumps using "JMP LBL" to a second "GET_OFFSET" which tries to load the vision process "DPGARAZ".

If neither "OFFSET" loads the appropriate process, the program skips to line 14 of the code, throws an error and jumps to the beginning of the program (the first line of code).

If "OFFSET" loads the vision process, it then calls the "DP_DUM" program or the "DP_GARAZ" program using the "CALL" command, depending on the type of vision process loaded.

4. Evaluation of Experiments

The evaluation is divided into 2 experiments (build and grasp) and a final overall evaluation. The aim of the experiments was to test the success/error rate of object assembly and grasping. The experiments were performed with re-gripping. In the case of re-gripping, the object always has a high score because it is placed directly at the center of the camera's field of view.

4.1. Experiment 1: Assembling the Objects into the Final Assembly

The aim of the experiment is to determine the quality of the assembly of objects on the base plate. Out of the 14 assembly tests (7 houses and 7 garages), there were two assembly errors shown in Table 1. Both errors were identical and occurred when the garage was assembled with object 1. The errors occurred on floor 4, where object 2 is also assembled (this is step 12). All tests were performed under 600 Lx illumination.

During testing, the platform was moved in the centre of the camera's field of view and with different rotations.

Table 1. Assembly errors in assembly tests

Build		
Total tests at 600 [Lx]		14
Final assembly	House	7
	Garage	7
Errors	House	0
	Garage	2
Errors [%]	House	0 %
	Garage	28,57 %
Overall system error rate		14,29 %

Corrective Actions

The step improvement was fixed by manually changing the coordinates of the already saved point where the gripper releases object 1.

A new point and therefore a new save position of object 1 was not possible, as the assembly was already offset in relation to the first "learning" objects and assembly, and thus the newly saved point would not fit the camera coordinate system.

After the implementation of the corrective action, no further compilation errors were found, and it can be confirmed that the corrective action significantly reduced the error rate of the system.

4.2. Experiment 2: Object Recognition and Grasping

The aim of the experiment is to determine the quality of vision and grasping processes in the field of view of the camera. The grasping was performed during the set-up and afterwards by testing individual objects in the field of view of the camera. The grasping during the assembly process recorded 8 errors, as shown in Table 2.

Table 2. Errors during the assembly process

Grasp			
Total amount of grasps		220	
Number of objects grasp	Object 1	90	
	Object 2	50	
	Object 3	90	
Errors	Object 1	1	1,1 %
	Object 2	3	6 %
	Object 3	4	4,4 %
Overall system error rate		8	3,69 %

Errors were recorded for all objects, and almost always the error was related to grasping the object at the edge of the camera's field of view. The scores for each object were also checked at each position in the field of view, results are in Table 3. The table represents quality scores for individual objects in each part of the camera's field of view.

Table 3. Quality scores for individual objects in each part of the camera's field of view

Quality scores for individual objects in each part of the camera's field of view				Object 1	Object 3	Not functional for objects 2 and 3
				Object 2	ALL Objects	
95,30%			100%			95%
	89%		92%		86%	
	98%		99%		94%	
	86%		99 - 100%		85%	
	98%		99%		97%	
	90%		89%		88%	
96,70%			99%			96,40%

Corrective Actions

Error on object 1, where object 1 was confused with object 2. On subsequent examination, it was found that object 1 could easily be confused with both object 2 and object 3 as it is part of these objects. The vision process was redesigned for object 1. The value of "Area Overlap" was reset to 0%. The "Area Overlap" function is a spatial numerical measurement that calculates the total area, length, or number of overlaps between features in the current layer and features in the target layer.

After the corrective action was taken, no further errors occurred for Object 1, and object sampling/grabbing was very reliable even at the edge of the camera field of view.

4.3. Overall Evaluation

The build did not show any other errors after correcting the error of saving object 1 to the build. The claim that the system error rate is 0% is not true, but the error rate is very close to 0%.

Testing the build was very challenging due to the non-functional automated program of the collaborative robot, so it was not possible to perform more tests. This evaluation shows the accuracy that can be achieved by combining robot and machine vision.

The grasping differs for each object. After fixing object 1, no errors were observed during the retrieval process or at the edge of the camera's field of view, allowing for relatively large possibilities of object placement on the retrieval surface. For object 2, errors mainly occurred due to placement at the edge of the camera's field of view, where the object was poorly recognized due to its coarse (gray) surface, resulting in no recognition of object 2 on the surface at all. For object 3, errors also occurred due to placement at the edge of the camera's field of view, leading to retrieval errors caused by poorer edge rendering and displacement of the central (retrieval) point. This evaluation demonstrates that the entire camera's field of view can be utilized. With more complex objects, the quality slightly deteriorates towards the edges of the field of view, as can be seen in Figure 13.

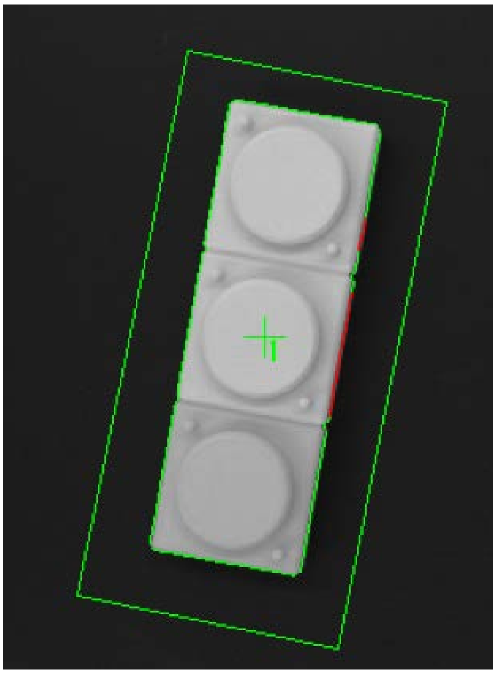


Figure 13. Object 3 with shifted centre and with visible deterioration of edge detection

5. Discussion

Out of a total of 8 errors, 4 were caused by improper grasping of object 3. The incorrect grasping consistently occurred at a similar retrieval position. Object 3, as shown in Figure 13, has a shifted center due to distortion, making it problematic for retrieval. One possible solution could be reducing the camera's field of view.

In the case where object 3 is not within the camera's field of view, the program would declare an error of "object 3 not found," and the camera would automatically adjust to a different capturing position.

From Figure 13, it can be observed that the centre identified through the vision process does not align precisely with the centre of the object. This difference results in issues with grasping.

During the "learning" phase, the object is taught that the gripper aligns precisely with the centre, and perfect grasping occurs when the centre is accurately detected. In this case, the centre is shifted, affecting the robot's approach accuracy. Due to the gripper type and overgripping, the robot can tolerate slight deviations in determining the object's centre. However, if the centre of the object is significantly displaced, the robot's gripper encounters difficulties in correctly grasping the object.

The other 3 errors were caused by not loading object 2. The error in not finding object 2 may be because object 2 had to be loaded more than once and the loading resulted in small differences in the individual objects 2 (3 pieces in total). Again, the errors were only at the edges of the camera field of view. The solution is to modify object 2 to make the objects identical.

In contrast to the previously mentioned papers [9], [10], [11], [13], the presented research investigates the error rate and accuracy of actual object grasping by a collaborative robot connected to a machine vision system versus pure machine vision evaluation.

In paper [12], the machine vision system is associated with a single-purpose system performing simple assembly. Compared to this research, the research we present focuses on a more complex device, i.e., a collaborative robot and a more complex assembly. Thus, the use of a collaborative robot instead of a single-purpose system is closer to the trends of the Industry 4.0 concept.

The advantage with comparison to related paper [14] is that our research is concentrating in the field industrial machine vision.

6. Conclusion

The research was developed at the Industry 4.0 laboratory at the Faculty of Mechanical Engineering of the University of West Bohemia in Pilsen. The aim of the work was to devise the logic of the experiment to achieve the highest success rate, to set up programs and vision processes so that everything works with the least number of errors and to test the success of the assembly and grasping of individual objects. While evaluating and testing the experiment, problems arose that were adjusted/corrected to benefit the success rate of the overall system.

In the practical part, the logic of the whole experiment (system) was described with the description of the individual objects, assemblies, tasks, and workplace. Subsequently, all vision processes were listed with a description of the vision process at the base and a description of the individual setup. In the programming section, basic commands were explained, the structure of the main program was shown, and flow charts of all programs were presented.

Flow charts were made to better understand the logic behind the creation of the programs. The linking and calling of programs are explained in the program hierarchy.

In summary, the theoretical part of the thesis describes machine vision systems and their use in cooperation with robots. The practical part focuses on the description and setup of basic movements, vision processes, programming of the robot and testing the success of the created experiment. Testing revealed some errors, which were described and eliminated at the end of the work.

In the following research, the research team's plan is to focus on testing grasping accuracy under different lighting conditions. The second area of planned research is then to acquire a 3D camera and compare the results presented in this paper with the results measured using the 3D camera.

Acknowledgements

This paper was created with the subsidy of the project SGS-2023-025 'Environmentally sustainable production' carried out with the support of the Internal Grant Agency of the University of West Bohemia.

References:

[1]. Daneshjo, N., Malega, P., Hlubeňová, J., & Štuller, P. (2022). Implementation of Simulation in the Design of Robotic Production Systems. *TEM Journal*, *11*(1), 179–188. Doi: 10.18421/TEM111-22

[2]. Hägele, M., Nilsson, K., Pires, J. N., & Bischoff, R. (2016). Industrial robotics. *Springer handbook of robotics*, 1385-1422.

[3]. Pollak, M., Kocisko, M., Paulisin, D., & Baron, P. (2020). Measurement of unidirectional pose accuracy and repeatability of the collaborative robot UR5. *Advances in Mechanical Engineering*, *12*(12). Doi: 10.1177/1687814020972893

[4]. Grznár, P., Gregor, M., Krajčovič, M., Mozol, Š., Schickerle, M., Vavrík, V., et al. (2020). Modeling and Simulation of Processes in a Factory of the Future. *Applied Sciences*, *10*(13). Doi: 10.3390/app10134503

[5]. Müller, J., Broum, T., & Malaga, M. (2022). USE OF MACHINE VISION SYSTEMS IN INDUSTRIAL ROBOTICS. *Engineering Sciences and Production Management*, 87-94. Slovakia: Pro Enviro.

[6]. Horejsi, P., Novikov, K., & Simon, M. (2020). A Smart Factory in a Smart City: Virtual and Augmented Reality in a Smart Assembly Line. *IEEE Access*, 8, 94330-94340. Doi: 10.1109/ACCESS.2020.2994650

[7]. Automate. (2014). *Computer Vision vs. Machine Vision*. Association for Advancing Automation. Retrieved from: https://www.automate.org/news/computer-vision-vs-machine-vision [accessed: 05 July 2023].

[8]. Robotics Tomorrow. (2019) *What is Machine Vision?* Robotics Tomorrow. Retrieved from: https://roboticstomorrow.com/article/2019/12/what-is-machine-vision/14548 [accessed: 12 June 2023].

[9]. Szűcs, B., & Ballagi, Á. (2019). Reducing Pseudo-error Rate of Industrial Machine Vision Systems with Machine Learning Methods. *Acta Technica Jaurinensis, 12*(4). Doi: 10.14513/actatechjaur.v12.n4.511

[10]. Yan, H., Achkar, A., Mishra, A., & Naik, K. (2016). Automated Failure Detection in Computer Vision Systems. *Journal of Computational Vision and Imaging Systems*, *2*(1). Doi: 10.15353/vsnl.v2i1.84

[11]. Huang, C-Y., Hong, J-H., & Huang, E. (2019). Developing a Machine Vision Inspection System for Electronics Failure Analysis. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, *9*(9), 1912-1925. Doi: 10.1109/TCPMT.2019.2924482

[12]. Chauhan, V., & Surgenor, B. (2015). A Comparative Study of Machine Vision Based Methods for Fault Detection in an Automated Assembly Machine. *Procedia Manufacturing*, *1*, 416-428. Doi: 10.1016/j.promfg.2015.09.051

[13]. Zhang, P., Wang, J., Farhadi, A., Hebert, M., & Parikh, D. (2014). Predicting Failures of Vision Systems. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 3566-3573. Doi: 10.1109/CVPR.2014.456

[14]. Erickson-Davis, C. (2023). What it is to see: Artificial vision as constitutive interaction. *Ethos*, 1-16. Doi: 10.1111/etho.12367

[15]. Fanuc. (2020). *CR-7iA and CR-7iA/L*. Fanuc. Retrieved from: https://www.fanuc.eu/cz/en/robots/robot-filter-page/collaborative-robots/collaborative-cr7ial [accessed: 16 June 2023].

[16]. Fanuc. (2020). *R-30iB Plus controller*. Fanuc. Retrieved from: https://www.fanuc.eu/cz/en/robots/accessories/robot-controller-and-connectivity [accessed: 16 June 2023].

[17]. Schunk. (2020). *Co-act EGP-C.* Schunk. Retrieved from: https://schunk.com/gb/en/gripping-systems/parallel-gripper/co-act-egp-c/c/PGR_3995 [accessed: 23 June 2023].

[18]. Zima, J. (2020). Programování průmyslových robotů FANUC. [*Programming of FANUC industrial robots]*. [Bachelor's thesis Faculty of Mechatronics, Informatics and Interdisciplinary Studies, Technical University of Liberec].