

Aaron Hortobágyi

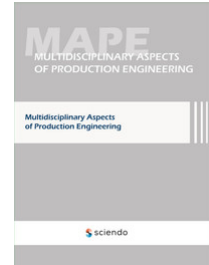
ORCID ID: 0000-0003-4017-7196

Technical University in Zvolen, **Slovak Republic**

Elena Pivarčiová

ORCID ID: 0000-0002-6676-8245

Technical University in Zvolen, **Slovak Republic**



INTRODUCTION

Robotic systems are currently one of the most important areas of innovation in production lines and production systems. The best example is the automotive industry. This currently defines the true nature of the word robot. But this is not the only sector where such modernization appears. We must realize that live in a time when the increasing scale of human labor is being taken over by robots. Development is progressing at fast pace and in the short term most of the current jobs may be held by robots and affect almost all areas of our lives.

As more and more professions are linked to modern technologies, demand of graduates of technical schools increases in the labor market. That is why today's generation of young people should be more guided in a technical direction, with focus on automation of different processes and on discovery of new technical devices, that would make life in the future easier.

Therefore, the Faculty of Technology of the Technical University in Zvolen focuses on the popularization of science and technology among young generation with emphasis to boost the interest of the young generation in the field of technology, try to attract the young generation to automation and robotics and to show them how to design and implement their own solutions.

This work is further focused on the transfer of graphical information to special software product. The product was a Lego Mindstorms Education EV3 set, which combines structural elements, servo motors, sensors, and a programmable unit. The kit has application possibilities in the introduction of the youth to the issue of robotics. It includes a block programming language, which can be quickly understood and adopted. Various studies were targeted at this topic already, aiming at basics of EV3 builds and programs (Rollins, 2014), (Bell, Kelly, 2017), (Harding, 2018), but also on more specific topics. Concerning the cooperation of EV3 with other platforms and coding environments, multiple articles were published about Arduino (Koch, 2020), android (Chen et al., 2018), Matlab Simulink (Klassner et al., 2012), (Lapusan et al., 2016), (Bevrnja et al.,

2019), (Montes et al., 2021), Python (Crnokić et al., 2020), CNL based program synthesis system (Zhan, Hsiao, 2018), and MakeCode as an environment for programming EV3 (Voštinár, 2020). Multiple authors addressed use of EV3 as introduction of students to robotics (Ponce et al., 2019), (Montés et al., 2019), (Matyushchenko et al., 2020), trajectory control problems (A. Kapitonov et al., 2018), and dynamic system control (Zhang, Wan, 2020). EV3 was also used as an Introductory platform for 2 Fuzzy Logic Controllers for Velocity Regulation (Valdez et al., 2021), observer-based reference-following feedback controller for a Segway robot (Hughes et al., 2020), gesture-based applications with use of Microsoft Kinect sensor (Dardan Maraj, Arianit Maraj, 2017).

Sensors bring a range of possibilities for interactions with the environment. The color recognition feature is enabled by the RGB sensor. However, this sensor is limited to only short distance from the subject. This research is focused on replacement of the mentioned sensor with a camera, and effect of replacement on a specific application for which a robot was built from the kit. The robot was designed to solve the Rubik's Cube. In this design, the RGB sensor was used to capture the color layout on the cube. The cube was gradually rotated, and the sensor was gradually moved above each tile, so the movement of the sensor was repeated 54 times. Our goal was to speed up color capturing with use of the camera and to implement communication and data transmission between smartphone and EV3. The ability of the camera to scan multiple tiles in a single shot was assumed as a possible advantage. This could reduce the time for cube rotation and speed up the solving process. Another possible advantage was possible improvement the reliability of the scan.

MATERIAL AND METHODOLOGY

Robotic kit

Lego Mindstorms Education is one of the most widely used robotic kits in the teaching process. It is suitable as an educational and motivational tool for students. By solving specific problem tasks involving several disciplines, students go through process of creative engineering and research work and get deeper understanding in the fields of informatics, automation, mechatronics, and robotics.

At the same time, these kits can be used for scientific and research purposes for teachers and students as the subject of bachelor's and diploma thesis.

The set offers various research possibilities, e.g., in the field of robotics, signal processing and real-time sensor data evaluation, space orientation and trajectory optimization.

The implementation and especially the use of robotic kits in the educational process includes active application of engineering analysis of problems, design, construction, programming, and testing students' own solutions. The interactive approach supports students' motivation, willingness to explore and discover, as well as formulation of hypotheses and solutions based on the development of their own abilities and the expansion of possibilities. Activities within educational

robotics develop the competences necessary for mastering the principles and design of the structure, as well as the subsequent construction of a functional robot, the use of sensors, collection, and interpretation of experimental data, understanding the principles of object movement, planning, testing, evaluation, and correction of errors in solutions.

With use of LEGO Mindstorms robots, students can break into the basics of decision-making and planning, processing of robot behavior in different situations, and use of elements of artificial intelligence. They can control the robot in real time, try parallel processing of more demanding tasks, or set multiple robots to communicate and collaborate simultaneously.

The interactivity of the learning process and the associated "learning by play", where students can immediately see the result of their work, leads them to active engagement in the solution of the given issue and thus to its faster understanding.

Educational robotics in the pedagogical process will allow students to engage imagination, analytical and synthetic thinking, develop a creative approach to the problem, find their own and original solution to problems of varying difficulty. Student cooperation in groups leads to the development of communication and to development of new working practices when working on group projects.

Robot MindCub3r

A robot model labeled as MindCub3r has been chosen for the research. The robot was constructed by David Gilday from Lego Mindstorms kit, with purpose to solve the Rubik's Cube. The original model is shown in Figure 1 (Gilday, 2014).

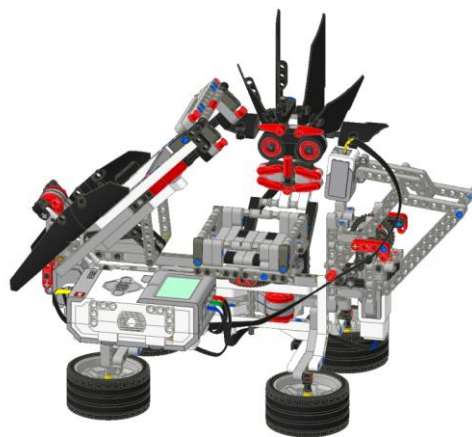


Fig. 1 Mindcub3r Rubik's cube solving robot

Source: (Gilday, 2014)

The software consists of three main parts (Gilday, 2014):

1. MindCub3r-v2p2.ev3 project file serving as the main program for motor and sensors control, programmed in original LEGO MINDSTORMS EV3 graphics environment

2. mc3solver-v2p2.rtf compiled in C++, containing an algorithm to find an effective sequence for cube solving
3. EV3 "MC3 Solver v2p2" application to run mc3solver-v2p2.rtf

The communication of the main program with the mc3solver-v2p2 algorithm.rtf is carried out through shared files on EV3.

For better color recognition, a modified program block was created for the sensor. Output of the block consists of RGB values (Gilday, 2014).

When launched, MindCub3r-v2p2.ev3 performs a scan to capture the colors of the tiles on the cube. If the data is insufficient, the scan is repeated twice more. After the third unsuccessful attempt, the loop is interrupted, and the process ends.

A scan consists of scanning individual walls and of the cube rotation. In original version scanning process involves top side of the cube. The cube is gradually rotated by 360°, while the RGB sensor is moved over individual tiles, reading corresponding RGB values.

After the scan completion, the scanned values were shared to mc3solver-v2p2.rtf, where the sequence of steps for cube solving was generated. This sequence was shared back to MindCub3r-v2p2.ev3 program, which controls the robot's motors.

The disadvantage of this approach was the relatively long time needed for scanning color of individual tiles, as the sensor must capture 54 values and cube has to be rotated 5 times in order for sensor to scan all faces. Another drawback was the fact that it was not possible to adjust the mc3solver-v2p2 section.rtf. In some cases, a scan error occurred, mostly due to change of ambient light during the scan and due to generally unsuitable light conditions (e.g., direct sunlight reflection from the cube).

To reduce the scan time and to increase the scan reliability, a camera was chosen. This brought a capability to scan the entire face of the cube in one shot. Another approach was the use of a smartphone, which could scan three sides from a single photo.

NXTcam V4 Camera

The NXTCam-v4 camera shown in Figure 2 was chosen as the first alternative.



Fig. 2 NXTCam-V4 Camera

Source: (Mindsensors, 2016)

The camera is compatible with NXT-G, EV3G, RobotC, LeJOS, NXC, LabVIEW and LVEE platforms. The sensor was directly designed as an extension for EV3 and NXT kits. Control blocks in the robot's original software were included with the product. The camera had two main purposes: line tracking and color-defined objects tracking. Camera could scan up to 30 frames per second. The camera focal length was adjustable, and lens had a built-in infrared light blocking filter (Mindsensors, 2016).

The camera enables position and size tracking of 8 distinct objects. The objects had to be defined by color in the additional NXTCamView program, shown in Fig. 3.

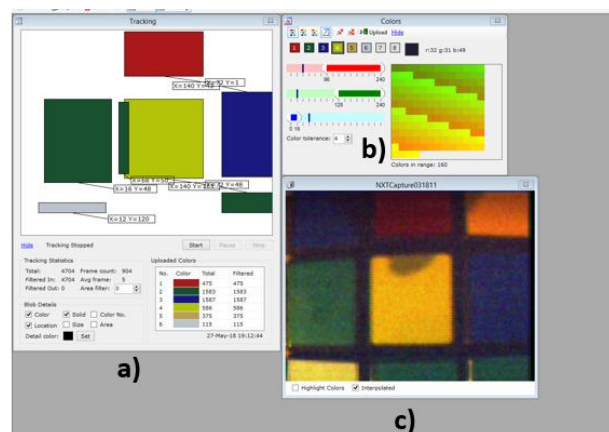


Fig. 3 Definition of objects for NXTCam v4 camera
 a) live output from the camera – display of objects with their coordinates,
 b) definition of objects by color, c) input image of the camera

Smartphone

As a second alternative, the selected extension was the smartphone camera. The disadvantage of this option was the necessity of a communication channel between the phone and the EV3 set. However, the multifunctionality of the device brought a great advantage – unlike a specific camera, there was no need for purchase of a new device. A great advantage of this option was the range of available programming environments and the additional accessories, which the phone was equipped with.

Samsung, LG, and Xiaomi smartphones were used as part of the project. For each type of device, it was necessary to fine-tune the program and the robot construction.

Robot construction

For both alternatives, the purpose of adjustment was to speed up the cube scanning process. The camera was set to capture multiple tiles on the cube at the same time. For NXTcam, process was modified to scan colors of nine tiles (the entire face of the cube) at one take. For the smartphone, approach of capturing of three faces of a cube (27 tiles) per shot was chosen, with smartphone placed in a static fixture. The motor that served for movement of

the RGB sensor and NXTcam was used to center the cube before the image creation. Modified structures for NXTcam and smartphone are shown in Fig. 4.



Fig. 4 Modification of the robot construction

- a) Construction for NXTcam, with camera fixed on a retracting arm
b) Construction for a robot with a phone fixture and a cube centering arm

Software

The phone app was created through the graphic environment Mit app Inventor 2. It consisted of three main blocks which can be described by their appropriate functions. To create a photo, a block developed by (Pura Vida Apps, 2010) was used. This block enables to take pictures programmatically and to adjust smartphone camera parameters, such as exposure length, focus, and usage of flash.

The communication between phone and EV3 was via EV3 Mailbox blocks, which were used to communicate through Bluetooth. The block was based on a description of the communication method via Bluetooth (Jerry, 2018).

Processing of the created image into usable data was carried out in a block created as a part of the study. The main steps the program goes through after taking photo is taken are shown in Figure 5.

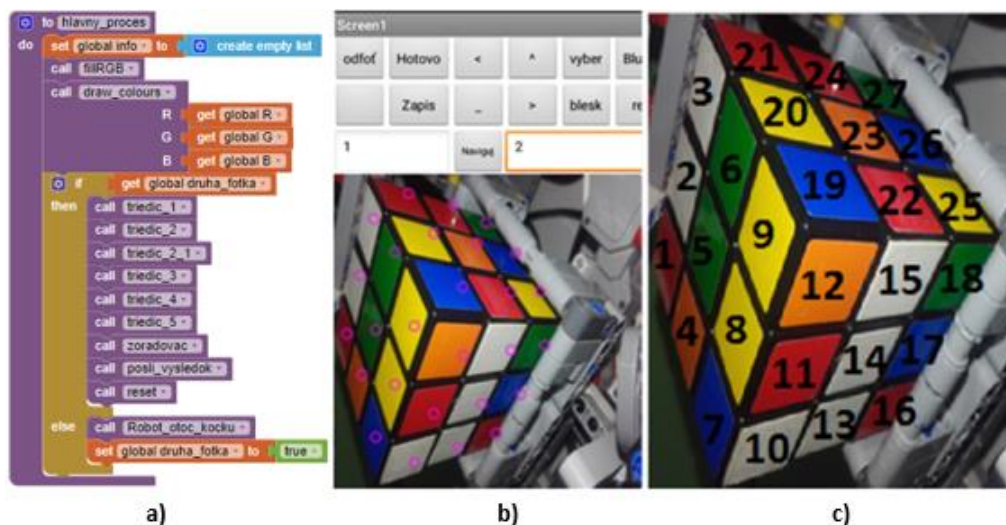


Fig. 5 Smartphone program

- a) main block of the program, b) reading of colors from the image,
c) color reading order

The principle of the program is based on taking picture of three faces of the cube at once. Thus, 27 sets of RGB values can be obtained from one picture. The order in which the colors were read is shown in Figure 5.

After cube rotation, another 27 values were read and saved.

After scanning the entire cube, information of 54 colors was sent to EV3 encoded as digits (white – 1, blue – 2, green – 3, yellow – 4, orange – 5, red – 6).

Transition of information about individual colors turned out to be too slow, as EV3 processing did not match speed of smartphone transmitter, and data loss occurred. Approach to put delay between messages would result in inappropriately long transfer time. Therefore, the color information was merged into nine six-digit numbers, so fewer messages would be sent. This format proved to be the most time efficient for convenient synchronization of the devices.

Before data transition, condition that each color was ascribed exactly to 9 tiles had to be met.

In the EV3 program, blocks for the cube rotation were modified. Additional blocks were created for data exchange with a smartphone, and for receiving of the scan output. Since the original input for the solving algorithm were RGB values captured by a light sensor on the robot, it was necessary to create a program section for adjustment of the required values and for change of order of values, to match the original format.

As the distance between the phone and the cube was considerably larger than in the original setup and multiple sides were scanned from one picture, the RGB values were in broader range when compared to use of the original sensor. Therefore, the RGB values captured by the phone were not considered as a suitable direct input for the solver algorithm and further minor adjustments had to be done.

RESULTS AND DISCUSSION

After robot construction and software were modified, cube scanning and solving took place. As the robots were designed for presentation of possibilities brought by automation, the scans had to be reliable under varying light conditions.

During solving process, the scan times were tracked. Results were summarized in Figure 6. Term “scan time” in the text describes time for acquisition of data necessary for a successful solving process. This would consist of one or multiple scans, with term “scan” defined as reading colors of all tiles on the cube. The NXTcam camera was set to capture the entire face of the cube, with output consisting of coordinates of objects representing individual tiles. After capturing nine objects, the cube was rotated, and scan was repeated on the next face. If the required number of objects was not captured, scan of the face was repeated until it became successful. If an error occurred in identification of the colors after scan of the whole cube, the entire scan process was repeated. After the third unsuccessful repetition, the process was automatically terminated.

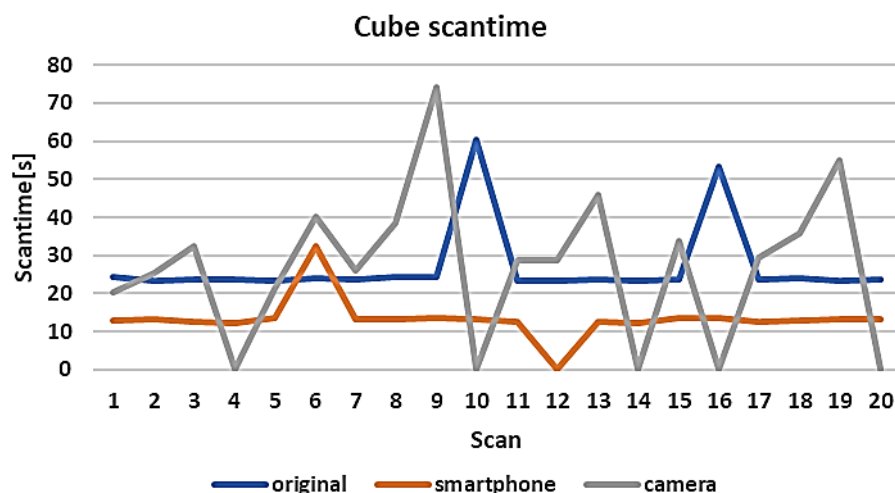


Fig. 6 Scan times of the original robot in a comparison with modified robots

In this scenario, the scan time has been represented in Figure 6. as a zero.

As seen in Figure 6, NXTcam approach resulted in the largest number of failed scans. In cases of successful scans, a large variance of scan times was observed. The obtained result was greatly influenced by light conditions. In controlled light, the camera showed no problem in object recognition by their defined colors. However, changes in shade and light reflection from of the cube walls led to inconsistent results.

The smartphone was set up to scan three faces of the cube at once, saving on time of cube rotation the between the scans. At the beginning of the scan, the robot pushed the cube into the corner of the stand, sending a message via Bluetooth. After receiving the message, the phone took a picture, and sent the robot a message to rotate the cube. RGB values of pixels with fixed positions were read from the images.

The disadvantage of the phone usage was in the limited communication speed between two devices. To get around this problem, there was an effort to keep the size of transferred data to a minimum. Therefore, an algorithm was created for the application to determine colors with additional verification of the number of tiles for each color.

The output from the application was in series of digits representing the colors on the cube.

The success of the scan was also influenced by type of ambient light. However, this problem turned out not too difficult to compensate for – the result was partially improved using a flash on the phone and by introduction of an algorithm for additional control of the result.

As seen in Figure 6, the results were much more consistent, when compared to the use of the NXTcam camera. The average scan time (13.3 s) was better than the average time achieved by the original robot (27 s).

In the second part of the experiment, the success of solving the cube was compared between the original robot and the robot modified with the smartphone.

The cube was scrambled into the same pattern in all experiments., A checkerboard pattern was used, in such a way, that the colors were in combinations that showed the largest error rate during color recognition tuning process. This pattern can be seen in Figure 7.

Three sets of experiments were conducted to verify the reliability of the proposed solutions:

- in natural daylight
- with addition of disruptive light source (illuminating upper face from a side)
- in artificial light (with no sunlight)

As part of the experiment, plan was to also monitor the ability of color recognition in environment without any light source. These conditions brought no significant result, due to an error in camera focus on the object. The error made it impossible to read the color from the correct positions. The effect of the error can be seen in Figure 7.

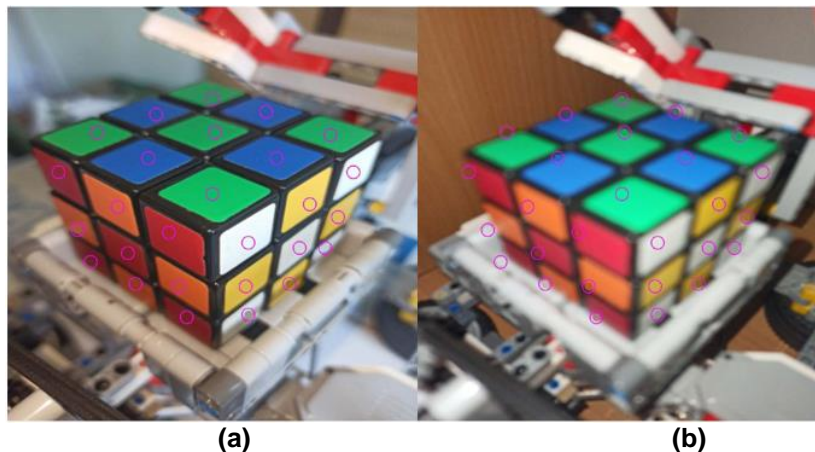


Fig. 7 Cube scrambled into checkerboard pattern
 a) Photo in normal operation
 b) Photo with the focus error

The reliability of the cube scan is shown in Figure 8. When the cube was solved after the first scan, a 100 % success rate was ascribed. When a scan repetition was needed, 50 % and a 33 % success rate was used for the third repetition.

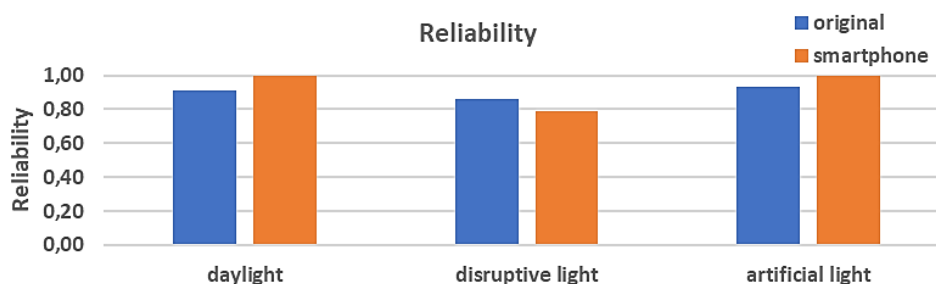


Fig. 8 Comparison of cube scan reliability1

As shown in Figure 8, both robots had reduced reliability in disruptive light.

In such case, second light source was added as a disruptive element, 20 cm from the edge of the cube from the side of the cube arm. In such assembly it was assumed that the reflection of light from the tiles could affect the accuracy of the recognition. In distractive light, the original robot achieved a reliability of 86 %, the designed robot using the smartphone achieved a reliability of 79 %. In disruptive light, smartphone scanning appears to be less reliable, and at least partial ambient light was needed to get a correctly focused image. In natural daylight, the original robot achieved a 91 % success rate. The robot with a smartphone achieved a 100 % success rate in a natural daylight. In artificial light, the original robot achieved a 93 % success rate of the solution, the robot with a smartphone achieved a 100 % success rate. Figure 9 shows the reliability of individual measurements with the disruptive light source positioned on the side of the cube.

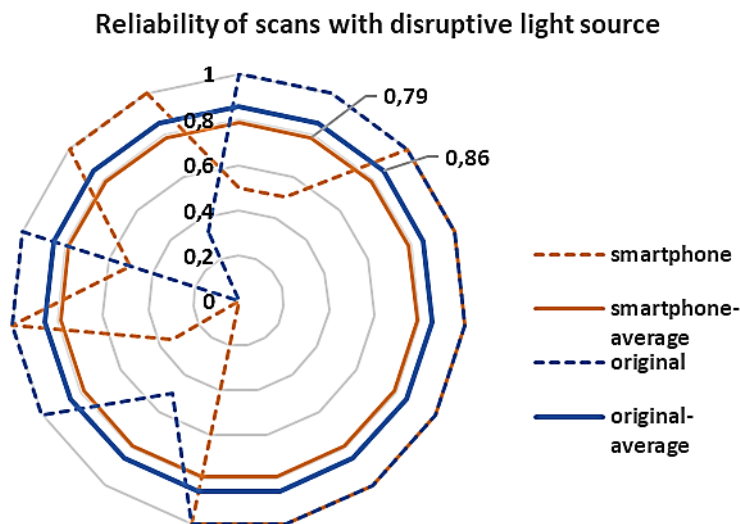


Fig. 9 Reliability of individual measurements in a disruptive light

In Fig. 10 scan times in daylight are shown. As seen in Figure 10, the scan time for colors is better for the smartphone variant in daylight.

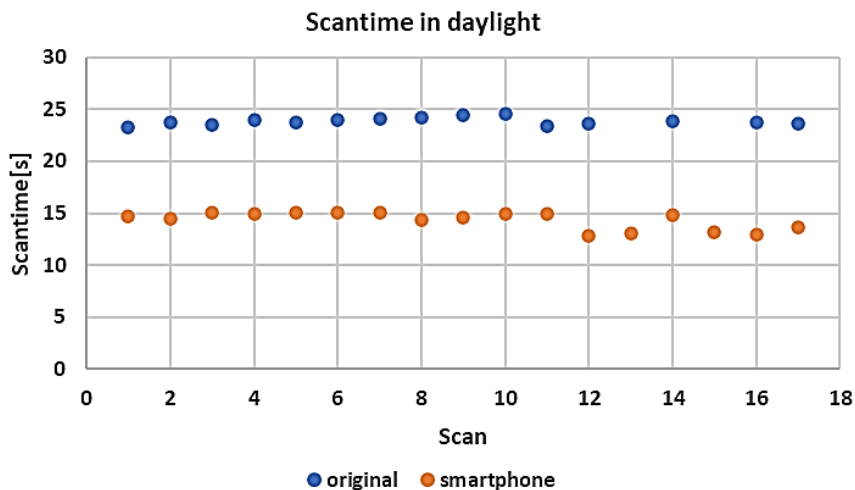


Fig. 10 Scan times in daylight

For both proposed modifications, communication with the target device was successful. Alternatives led to different results in the app: While the phone proved to be a suitable alternative, the NXTcam camera turned out as not appropriate solution. Although the use of the camera led to results that would be considered as positive in the controlled environment (scan time less than 24 s), the error rate was too large in real conditions. The use of a smartphone brought noticeable improvements, but it brought necessity to get familiar with a new programming environment. Another drawback was the need to replace the smartphone with another model, in such case it was necessary to modify part of the code.

Use of a smartphone for color recognition has proven to be a good alternative, color scans resulted as more reliable in standard daylight and artificial evening light, which are common lighting conditions for promo actions. Smartphone implementation brought worse results when in combination with disruptive light and was found to be not suitable for complete darkness, as a focus error appeared in such cases (Fig. 7). resulting in reduced the success of color recognition.

CONCLUSION

Robotics and automation currently have an important support role in the manufacturing industry. Sensors are an integral part of this task, due to their ability to obtain control data for robotic and automation systems. During the introduction of new sensors, many problems occur, such as measurement, calibration, data collection, data processing, filtering of data errors, etc. (Božek, 2019).

For modern teaching of robotics, algorithm creation and programming, it is necessary to introduce activities that will motivate students, attract interest in robotics, algorithm creation and programming, develop the skills necessary for the development and research work and active group cooperation.

Therefore, it is necessary to inspire interest of the youth in the robotics. To show that robotics is not only a theory, but also an interesting and creative discipline where enthusiasts can design, construct and program anything.

We are trying to develop ideas, projects and initiatives that would make the technique interesting and meaningful. We hope that we will be able to highlight the interest of the younger generation in the study of technology. Such ideas include the improvement of the Rubik's Cube Robot, which is currently used in promo events. Our goal was to speed up color scanning sequence with use the camera and establish communication and data transmission between smartphone and EV3.

From the results of this research, it can be concluded that the NXTcam camera was not an ideal solution for the application. Although use of this camera did not require attention for communication with EV3 and it used the same graphical environment as EV3, the use of this camera could achieve a much better result

in applications where the robot is supposed to move in the environment or needs to recognize objects and determine their position.

Smartphone usage brought the desired result: increased scan reliability and faster scanning process. The introduction of this alternative required learning of basics of the new environment, but the graphical form of the mit app inventory allowed almost intuitive use. The disadvantage of this alternative was the relatively complex establishment of communication channel with EV3. This was achieved by modification of the procedure, described by (Jerry, 2018).

With results stated, there is an assumption that additional optimization of the model with the smartphone could speed up the scanning process even more.

Incorporating collaboration of EV3 with sensors on the smartphone demonstrates potential for other applications, whether as a tracking tool, engine control (appinventor, 2012), or as a device for entering input data (Jerry, 2018).

ACKNOWLEDGEMENT

This contribution was prepared within the framework of the project KEGA 006STU-4/2021: „Progressive form of interdisciplinary education and support for the development of the study of vocational subjects in the university environment“.

LITEATURE

- Bell, M., Kelly, J. (2017). LEGO® MINDSTORMS® EV3. Berkeley: Apress. 333 p. ISBN 978-1-4842-2261-4
- Bevrnja, F., Bevrnja, M., Petrovic, M. (2019). Dynamic analysis, modeling and control of the LEGO EV3 modular mobile platform. Periodicals of Engineering and Natural Sciences. [online] Vol. 7(4), pp. 1868-1878. Available at: <http://pen.ius.edu.ba/index.php/pen/article/view/1084> [Accessed 25 Apr. 2021]
- Božek, P. (2019). Nonlinear modeling of the arm dynamics for the manipulators with reverse sensory validation. International Journal of Advanced Robotic Systems 16(6). [online] Available at: <https://journals.sagepub.com/doi/10.1177/1729881419889546> [Accessed 10. Apr. 2021]
- Crnokić, B., Pehar, F., Spajić, J. (2020). STEM Classroom: Creating a Python Application for an EV3 Brick Robotic System Used to Transport 3D Printed Boxes. In: Proceedings of the 31st International DAAAM Virtual Symposium "Intelligent Manufacturing & Automation". [online] Vienna: DAAAM International, pp. 88-97. Available at: https://www.daaam.info/Downloads/Pdfs/proceedings/proceedings_2020/012.pdf [Accessed 29 Apr. 2021]
- Gilday, D. (2013). How to build MindCub3r for LEGO MINDSTORMS EV3. [online] Available at: <http://mindcuber.com/mindcub3r/mindcub3r.html> [Accessed 15 Mar. 2021]
- Harding G. (2018). Programming LEGO® EV3 My Blocks. Berkely: Apress. 136 p. ISBN 978-1-4842-3437-2

- Hortobágyi, Á. (2018). Kamery a ich využitie v priemysle: Bakalárska práca. Zvolen: Technická univerzita vo Zvolene. Fakulta environmentálnej a výrobnjej techniky. 36 p.
- Hughes, T., Willetts, G., Kryczka, A. (2020). LQG controller for the LEGO MINDSTORMS EV3 Gyroboy Segway robot. IFAC-PapersOnLine. [online] Volume 53(2), pp. 17282-17287. Available at: <https://doi.org/10.1016/j.ifacol.2020.12.1811> [Accessed 3 May. 2021]
- Chen, Q., Chen, Y., Tang, P., Chen, R., Jiang, Z., Deng, A. (2018). Indoor Simultaneous Localization and Mapping for Lego Ev3. In: International Conference on Computer, Communication and Network Technology (CCNT 2018). [online] Lancaster: DEStech Publications, Inc., pp. 500-504. Available at: <http://dpi-proceedings.com/index.php/dtcse/article/view/24749> [Accessed 6 May. 2021]
- Jerry. (2018). Further AI2/EV3 Bluetooth Coding [online] Jander's LEGO® Stuff. Available at: <https://r.jander.me.uk/index.php/2018/03/10/further-ai2-ev3-bluetooth-coding/> [Accessed 22. Mar. 2021]
- Kapitonov, A., et al., (2018). Lego Mindstorms EV3 for teaching the basics of trajectory control problems. In: 2018 IEEE Frontiers in Education Conference (FIE) [online], IEEE, pp. 1-4. Available at: <https://ieeexplore.ieee.org/document/8659322> [Accessed 10 May. 2021]
- Klassner, F., Peyton-Jones, J.C., Lehmer, K. (2012). Genetic Algorithms with Lego Mindstorms and Matlab In: Proceedings Of The Twenty-Fifth International Florida Artificial Intelligence Research Society Conference. [online] California, Palo Alto: AAAI Press. Available at: <http://www.csc.villanova.edu/~klassner/pubs/FLAIRS2012-KlassnerPeytonJonesLehmer.pdf> [Accessed 11 May 2021]
- Koch, G. (2020). The LEGO Arduino Cookbook. Berkey: Apress. 234 p. ISBN 978-1-4842-6302-0
- Lapusan, C., Hancu, O., Rad, C., Dache, L., Maties, V. (2016). Integrated learning platform based on lego NXT and Matlab for teaching mechatronics. In: 8th International Conference on Electronics, Computers and Artificial Intelligence (ECAI). [online] Yekaterinburg: IEEE, pp. 1-4. Available at: <https://ieeexplore.ieee.org/document/7861145> [Accessed 5 May. 2021]
- Maraj, D., Maraj, A. (2017). Implementation of gesture based applications and communication with Lego Mindstorm EV3, In: 2017 6th Mediterranean Conference on Embedded Computing (MECO) [online], IEEE, pp. 1-4. Available at: <https://ieeexplore.ieee.org/document/7977236> [Accessed 9 May. 2021]
- Matyushchenko, I., Zvereva, E., Lavina, T. (2020). Development of Algorithmic Thinking by Means of Lego Mindstorms Ev3 on Robotics. In: Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBREIT). [online] Yekaterinburg: IEEE, pp. 444-447. Available at: <https://ieeexplore.ieee.org/document/9117764> [Accessed 7 May. 2021]
- Mindsensors collective. Vision Subsystem (2016). Camera for NXT or EV3 (NXTCam-v4). 2018 [online]. Available at: <http://www.mindsensors.com/ev3-and-nxt/14-vision-subsystem-camera-for-nxt-or-ev3-nxtcam-v4> [Accessed 5. Jan. 2021]
- Montes, N., Rosillo, N., Mora, MC., Hilario, L. (2021) A Novel Real-Time MATLAB/Simulink/LEGO EV3 Platform for Academic Use in Robotics and Computer Science. Sensors, [online] Volume 21(3), p. 1006. Available at: <https://doi.org/10.3390/s21031006> [Accessed 5 May. 2021]

- Montés, N., Rosillo, N., Mora, M., C., Hilario, L. (2019). Real-Time Matlab-Simulink-Lego EV3 Framework for Teaching Robotics Subjects. In: Lepuschitz W., Merdan M., Koppensteiner G., Balogh R., Obdržálek D. (eds) Robotics in Education. RIE 2018. Advances in Intelligent Systems and Computing, vol 829. Springer, Cham. [online] pp. 230–240. Available at: https://doi.org/10.1007/978-3-319-97085-1_23 [Accessed 2 May. 2021]
- Ponce, P., Molina, A., Mata, O., Baltazar, G. (2019). LEGO® EV3 Platform for STEM Education in Elementary School. In: ICEIT 2019: Proceedings of the 2019 8th International Conference on Educational and Information Technology. [online] New York: Association for Computing Machinery, pp. 177-184. Available at: <https://dl.acm.org/doi/10.1145/3318396.3318426> [Accessed 4 May. 2021]
- Pura Vida Apps. (2017). Camera Extension [online]. App Inventor Extensions Available at: <https://puravidaapps.com/camera.php> [Accessed 22. Mar. 2021]
- Rollins, M. (2014). Beginning LEGO MINDSTORMS EV3., Berkeley: Apress. 280 p. ISBN 978-1-4302-6436-1
- Valdez, F., Castillo, O., Caraveo, C., Peraza, C. (2019). Comparative Study of the Conventional Mathematical and Fuzzy Logic Controllers for Velocity Regulation. Axioms. [online] Volume 8(2), p. 53. Available at: <https://doi.org/10.3390/axioms8020053> [Accessed 9 May. 2021]
- Voštinár, P., (2020). MakeCode for Lego Mindstorms EV3. International Journal of Online and Biomedical Engineering, [online] Volume 16(4), pp. 42-53. Available at: <https://online-journals.org/index.php/i-joe/article/view/17069> [Accessed 2 May. 2021]
- Zhan, Y., Hsiao, M., S. (2018). A Natural Language Programming Application for Lego Mindstorms EV3. In: 2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR). [online] IEEE, pp. 27-34. Available at: <https://ieeexplore.ieee.org/document/8613631> [Accessed 29 Apr. 2021]
- Zhang, M., Wan, Y. (2020). Improving learning experiences using LEGO Mindstorms EV3 robots in control systems course. The International Journal of Electrical Engineering & Education, [online]. Available at: <https://doi.org/10.1177/0020720920965873> [Accessed 4 May. 2021]
- appinventor.mit.edu.(2012). Mit App inventor's official website. [online] Available at: <http://appinventor.mit.edu/> [Accessed 7. Apr. 2021]

Abstract: This article is focused on enhancement of HW/SW device by cooperation with a smartphone interface. The device was a programmable Lego Mindstorms Education EV3 set in the form of a robot designed to solve the Rubik's Cube. The aim of the research was to replace the built-in color sensor with a camera that would allow the cube scanning process to be accelerated. Two approaches were chosen to meet the goal: the NXTcam camera, accessible as an accessory to expand the set, and the camera built into the smartphone. The use of NXTcam led to better scan time, but this result was prone to external influences. The camera on the smartphone sped up the scanning process to 57% of the original time. The impact of external factors on the outcome was significantly lower, compared to NXTcam. In the experiment, the cube solving process was observed in natural light, with addition distractive light source and in artificial light.

Keywords: Graphic Information Transfer, Lego Mindstorms Education