

A SET OF REGRESSION MODELS FOR CONTROLLING A TROLLEY EQUIPPED WITH A ROBOT MANIPULATOR INTEGRATED INTO THE TECHNOLOGICAL PROCESS

I.Yu. Kholodilin¹, kholodilini@susu.ru
M.A. Grigorev¹, grigorevma@susu.ru
V.A. Kushnarev¹, a55538a@yandex.ru
N.V. Savosteenko¹, savosteenkonv@susu.ru
D.V. Spitsin¹, yanda.32@mail.ru
O.I. Osipov², osipovoi2011@yandex.ru

¹ South Ural State University, Chelyabinsk, Russia

² National Research University "Moscow Power Engineering Institute", Moscow, Russia

Abstract. The paper focuses on the problem of virtual environment systems and mobile robot simulators. The simulator utilizes the Robotics System Toolbox and Robotics Toolbox which help it to enhance pedagogical experience of computer science courses. Thanks to the mobile robot equipped with the fisheye camera and structured light, this simulator, created using Unity, can be used in learning navigation techniques taking place inside the indoor environment. An interactive indoor environment with different obstacles is also included in the virtual environment. The main goal of the simulator is to motivate students to keep studying robotics and consequently improve the quality of engineering education. We believe that integrating these tools into the educational process will increase students' interest in the subject and as a result, students will get a valuable practical experience. This paper not only demonstrates the compatibility of the proposed simulator with Matlab toolboxes, but also presents comparison analysis between considered toolboxes.

Keywords: virtual environment systems, mobile robot simulators, robotics system toolbox, robotics toolbox, navigation techniques, pedagogical experience, simulation platform

Acknowledgments. The study was supported by the Russian Science Foundation grant No. 24-29-20196, <https://rscf.ru/project/24-29-20196>.

For citation: Kholodilin I.Yu., Grigorev M.A., Kushnarev V.A., Savosteenko N.V., Spitsin D.V. A set of regression models for controlling a trolley equipped with a robot manipulator integrated into the technological process. *Bulletin of the South Ural State University. Ser. Power Engineering*. 2024;24(4):47–55. DOI: 10.14529/power240406

Научная статья
УДК 68-83
DOI: 10.14529/power240406

НАБОР РЕГРЕССИОННЫХ МОДЕЛЕЙ ДЛЯ УПРАВЛЕНИЯ ТЕЛЕЖКОЙ, ОСНАЩЕННОЙ МАНИПУЛЯТОРОМ РОБОТА, ИНТЕГРИРОВАННЫМ В ТЕХНОЛОГИЧЕСКИЙ ПРОЦЕСС

И.Ю. Холодили́н¹, kholodilini@susu.ru
М.А. Григорьев¹, grigorevma@susu.ru
В.А. Кушнарёв¹, a55538a@yandex.ru
Н.В. Савостеенко¹, savosteenkonv@susu.ru
Д.В. Спицин¹, yanda.32@mail.ru
О.И. Осипов², osipovoi2011@yandex.ru

¹ Южно-Уральский государственный университет, Челябинск, Россия

² Национальный исследовательский университет «МЭИ», Москва, Россия

Аннотация. В данной статье рассматривается проблема систем виртуальной среды и симуляторов мобильных роботов. В симуляторе используется Robotics System Toolbox и Robotics Toolbox для повышения эффективности преподавания курсов информатики. Благодаря мобильному роботу, оснащённому камерой «рыбий глаз» и структурированным светом, этот симулятор, созданный с использованием Unity, можно использовать для обучения навыкам навигации в помещении. В виртуальную среду также включена интерактивная внутренняя среда с различными препятствиями. Основная цель тренажёра – мотивировать студентов продолжать изучать робототехнику и, следовательно, повысить качество инженерного образования. Мы считаем, что интеграция этих инструментов в образовательный процесс повысит интерес студентов к предмету и, как следствие,

студенты получают ценный практический опыт. В данной статье не только демонстрируется совместимость предлагаемого симулятора с наборами инструментов MATLAB, но и проводится сравнительный анализ между рассмотренными наборами инструментов.

Ключевые слова: системы виртуальной среды, симуляторы мобильных роботов, набор инструментов для робототехнических систем, набор инструментов для робототехники, методы навигации, педагогический опыт, платформа для моделирования

Благодарности. Исследование выполнено при поддержке гранта Российского научного фонда № 24-29-20196, <https://rscf.ru/project/24-29-20196>.

Для цитирования: Kholodilin I.Yu., Grigorev M.A., Kushnarev V.A., Savosteenko N.V., Spitsin D.V. A set of regression models for controlling a trolley equipped with a robot manipulator integrated into the technological process // Вестник ЮУрГУ. Серия «Энергетика». 2024. Т. 24, № 4. С. 47–55. DOI: 10.14529/power240406

Introduction

During the last decades, we observe a huge increase in interest towards engineering and computer science courses, e.g. electrical engineering, electronic engineering and mechanical majored courses. More and more students choose this direction for their future career. At the same time, mobile robots are slowly getting to be adopted for those courses. For example, in some universities students are able to acquire a valuable experience and useful skills by working with real robots [1]. Teamwork and problem-solving are important aspects towards flexibility of the educational process, as it allows students to arouse their curiosity by exploring practical tasks. However, some educational centers may suffer from the lack of real robots. Consequently, various online platforms where students are able to acquire a variety of practical skills in the field of engineering and computer science are becoming popular [2]. These trends make the issue of teaching methods for the following sciences more relevant. It is also worthwhile mentioning that students do not always have enough motivation and involvement in the educational process; as a result, we do not always observe their high academic performance in those fields. Teaching materials which are presented in a simple form and are overloaded with diagrams and text may cause those problems defined earlier. Secondly, these standard teaching materials make the educational process repetitious. Thus, students often lack real, practical experience related to the disciplines they are studying. Standard educational process primarily gives an understanding of how the computer and engineering knowledge can be applied to science, business, industry, and other fields only theoretically. In standard programming or math classes, students usually solve various problems that are poorly related to their future profession.

The educational process has been broadly discussed by scholars [3–6]. The literature analysis demonstrates that the most appropriate way to combine theory with practice under current situation and within trends defined earlier is the use of simulation platforms. Moreover, virtual technologies are found to be more attractive for students than traditional educational materials [7]. MATLAB is a widespread software for teaching engineering and computer science courses. MATLAB is an interactive environment for pro-

gramming, data analysis and for the development and visualization of algorithms. An important feature of this program is the possibility of working with a variety of toolboxes. These toolboxes are able to simulate practical tasks, e.g. to model and program mobile robots. The Robotics System Toolbox [8] developed by a MATLAB team or the Robotics Toolbox [9] developed by Peter Corke can be named as a compact education system with tutorials and educational materials. These toolboxes have the following capabilities: planning the trajectory of mobile robots, creating algorithms, localization, monitoring the trajectories and, in general, controlling the mobile robot. The main drawback of them is the lack of realistic graphics and an appropriate interaction with objects on the scene. The study on the challenges of using robotics simulators for testing conducted in [10] also stated that the general challenges include the difficulties of learning and using simulators, lack of realism, and the absence of specific capabilities.

In the view of the current problems and trends, this paper presents an extension which can be used for studying mobile robots in combination with the Robotics System Toolbox and Robotics Toolbox. The proposed simulation environment in our previous study [11] showed how helpful it can be in a way of conducting experiments before testing systems in real environments. Here, we decided to move further and implement communication between MATLAB and our simulator by TCP/IP. In doing so, users familiar with the Robotics System Toolbox and Robotics Toolbox with the aid of our extension will be able to get a realistic rendering mode and interactive scenarios for operating the mobile robot equipped with the fisheye camera and a structured light. Moreover, we will conduct a comparison analysis as well as discuss the main features and compatibilities.

Extension capabilities

The outcome of adding robotics toolboxes to the MATLAB provides several convenient sets of robotics tools and opens new ways for better understanding and analyzing theoretical foundations of robotics and computer vision. The MATLAB environment is equipped with really convenient and simple modeling and visualization features. However, there are several drawbacks as well, e.g. the lack of photo-

realism of the virtual environment, which can interfere with visualization tasks. Game engines, such as Unity, provide more flexibility if we want to visualize a variety of scenes. Secondly, it is also important to know how to process data from real sensors of the vision system. A variety of publicly available datasets can be used for further processing by MATLAB. However, external virtual environments can help not only to collect data from sensors, but also provide more flexibility for users in generating their own datasets. In addition, from the point of view of learning engineering disciplines, they can be considered as an effective transition from theory to practice. This leads to an important thesis: if we can expand the set of visualization tools for systems such as MATLAB toolboxes, we will be able to solve a number of pedagogical problems and give engineering education several new useful insights. By taking this information into consideration, we decided to propose a photo-realistic extension for MATLAB toolboxes. We will demonstrate the compatibility of our extension with two case studies, which are based on the Robotics System Toolbox and Robo-

tics Toolbox. Our simulator is built by the Unity. The main screen of the simulator is presented in Fig. 1.

The main parts of the scene consist of the mobile robot, laser emitter, and fisheye camera; these objects can be controlled by the corresponding buttons located on the panels. This is a cross-platform simulator as it can be installed on Windows, macOS, and Linux operating systems. The features of the simulator are three-fold: (1) interaction with objects within a virtual environment, (2) communication with other programs via TCP/IP, (3) technical support and user guide. This functionality can be helpful for testing theories and modelling systems, while operating with types of sensors was not included to the Robotics System Toolbox and Robotics Toolbox before.

The simulator supports two modes: manual mode and automatic mode, which represents communication with other software via TCP/IP. In this paper, we consider navigation inside the indoor environment based on the MATLAB toolboxes and proposed extension for performing these programs. The required functions are presented in Table 1.

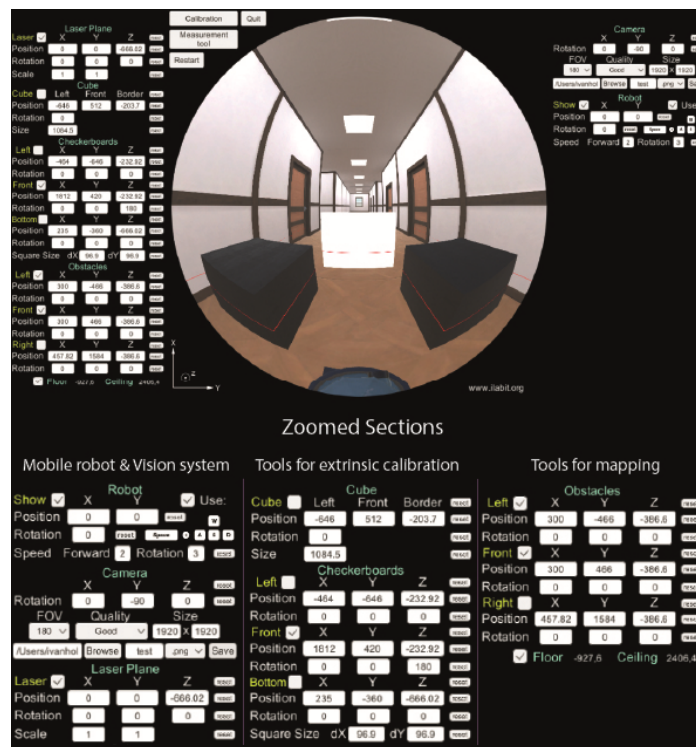


Fig. 1. Main screen of the proposed extension

Table 1

Operating modes of the simulator

Action	Manual mode from the user interface (UI) of the simulator	Communication with other programs by TCP/IP
Image capturing	Image can be saved with the corresponding button. Section “Camera” (see Fig. 1)	Image can be obtained in MATLAB via TCP/IP
Movements of the mobile robot	Mobile robot can be controlled by keyboard or input fields. Section “Robot” (see Fig. 1)	Mobile robot can be controlled in MATLAB via TCP/IP
Laser activation	Laser can be controlled by section “Laser Plane”	Laser can be controlled in MATLAB via TCP/IP

Map transform and regression models

System Model. This section introduces the system model and considers the process of obtaining a map with its further transforming to the formats supported by the Robotics Toolbox and Robotics System Toolbox for navigation of the mobile robot. Mapping is possible with the aid of the fisheye camera and laser emitter included in the simulation environment. It is worthwhile mentioning that we previously described the system model based on our previous work [11] and therefore provide just brief explanations in this section. World coordinates of the laser plane (X, Y, Z) can be obtained by the following equation:

$$\begin{bmatrix} u \\ v \\ f(\rho) \end{bmatrix} \times [r_1^c \ r_2^c \ r_3^c] [r_1^l \ r_2^l \ r_3^l \ t^l] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = 0, \quad (1)$$

where pixel coordinates of the image are represented by u and v ; relationship between world coordinates and image points is represented by extrinsic parameters, namely by column vectors r^c of the camera matrix and vectors r^l, t^l of the transformation matrix of the laser plane; The polynomial $f(\rho)$ is a part of the intrinsic parameters. For more information about the parameters see explanation in [11, 12].

The distance between the camera and laser emitter is constant. This means that for measurements the coordinates belonging to obstacles are changing along axes Y, Z and the distance from the camera to laser plane along X -axis does not change. The last one is represented by the 1st row of the column vector. Knowing that only (Y, Z) coordinates are changing, we can transform (1) into:

$$\begin{bmatrix} u \\ v \\ f(\rho) \end{bmatrix} \times [r_1^c \ r_2^c \ r_3^c] [r_1^l \ r_2^l \ t^l] \begin{bmatrix} Y \\ Z \\ 1 \end{bmatrix} = 0. \quad (2)$$

Map Transform and Regression Models. Knowing intrinsic and extrinsic parameters of the vision system, the image containing a laser beam belonging to obstacles can be transformed to the map. First of all, the laser beam must be detected on the fisheye image and extracted from it. This procedure can be carried out by thresholding [13] as the laser beam has its unique red color. Once the laser beam was extracted, we can transform laser points on the image to the world coordinates by (2). The transformed coordinates are represented by blue dots (Fig. 2). The initial map presented in Fig. 2 contains negative coordinates of obstacles which cannot be transformed to the binary map of the Robotics System Toolbox and to the logical map of the Robotics Toolbox. Thus, the initial map has to be shifted for eliminating those negative coordinates. Shifted coordinates can be obtained by (3) and (4) for coordinates along X - and Y -axes.

$$X_{sifted} = X_i + X_{min}; \quad (3)$$

$$Y_{sifted} = Y_i + Y_{min}, \quad (4)$$

where X_{sifted} and Y_{sifted} represent shifted coordinates; X_i and Y_i represent current coordinates; X_{min} and Y_{min} represent the minimum coordinates. The shifted map is shown in Fig. 3.

Once the shifted map with positive coordinates was created, it can be converted to the maps for corresponding toolboxes. Firstly, the shifted map is converted to the format suitable for the Robotics System

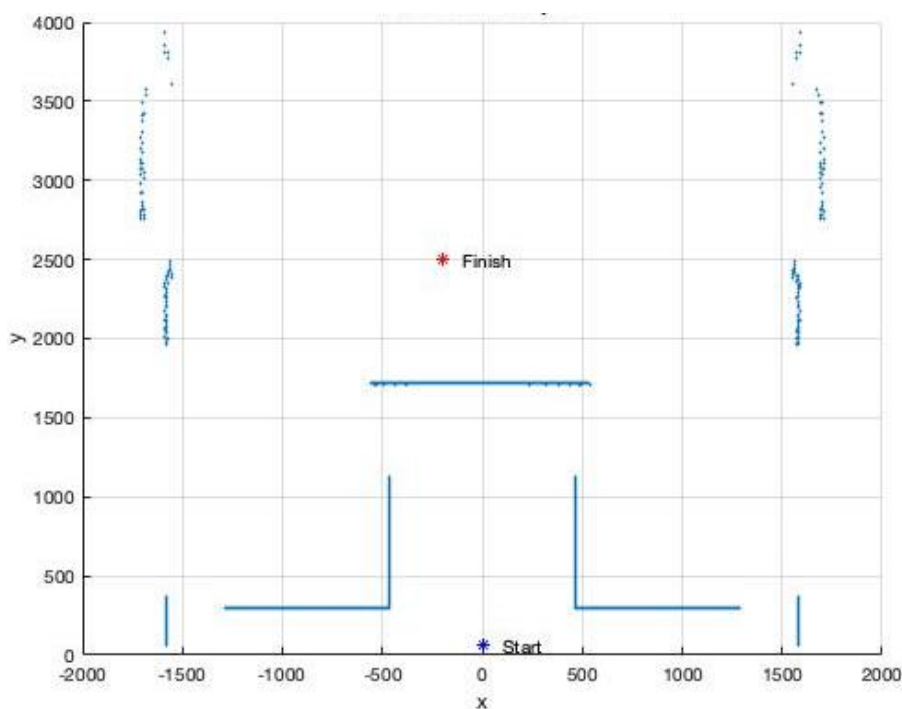


Fig. 2. Initial map obtained with a laser emitter

Toolbox, namely to the binary map with the aid of the MATLAB function “binaryOccupancyMap”. The binary map is shown in Fig. 4. Next, the coordinates of the binary map can be simply converted to

the format suitable for the Robotics Toolbox, namely to the logical map with the aid of the MATLAB function “occupancyMatrix”. The logical map is shown in Fig. 5.

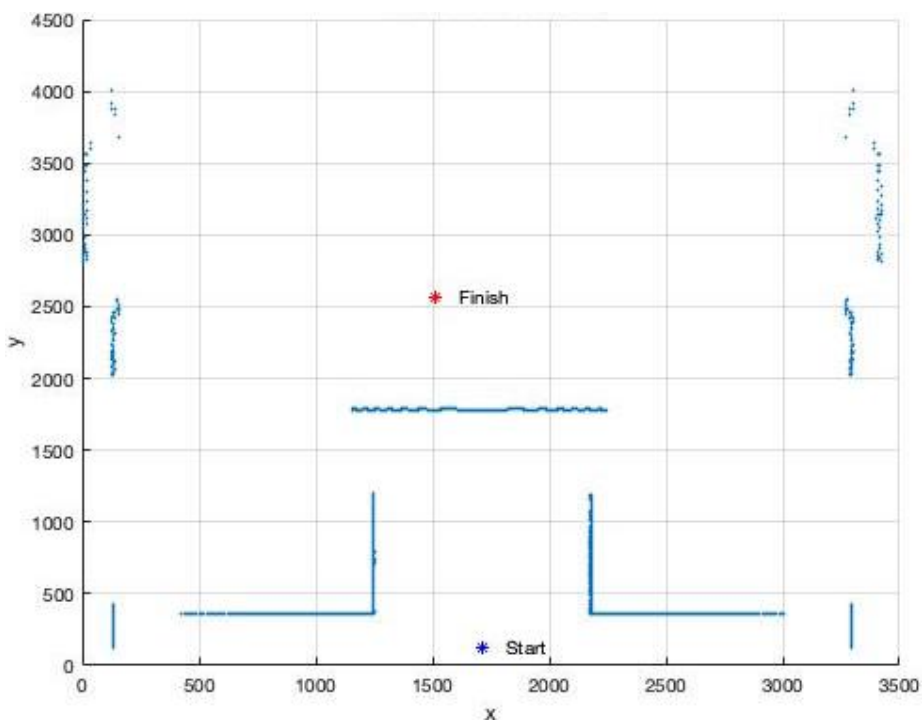


Fig. 3. Shifted map obtained

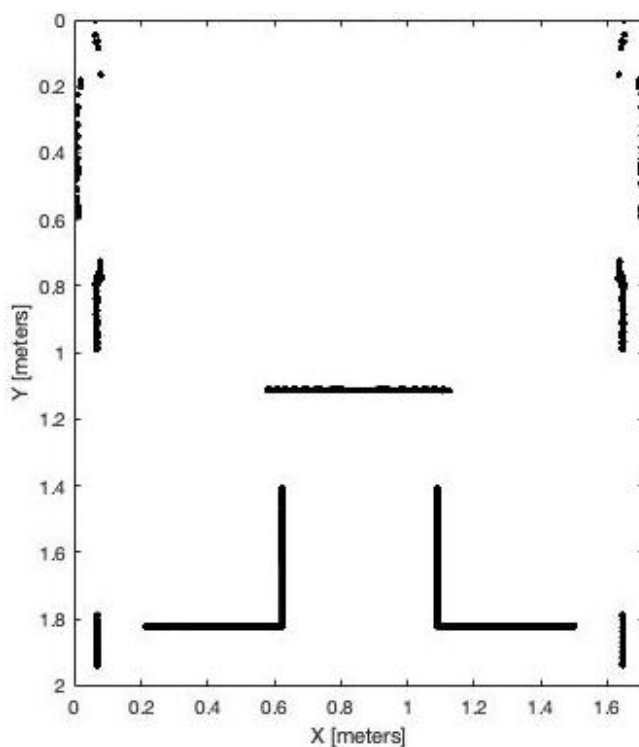


Fig. 4. Binary map supported by the Robotics System Toolbox

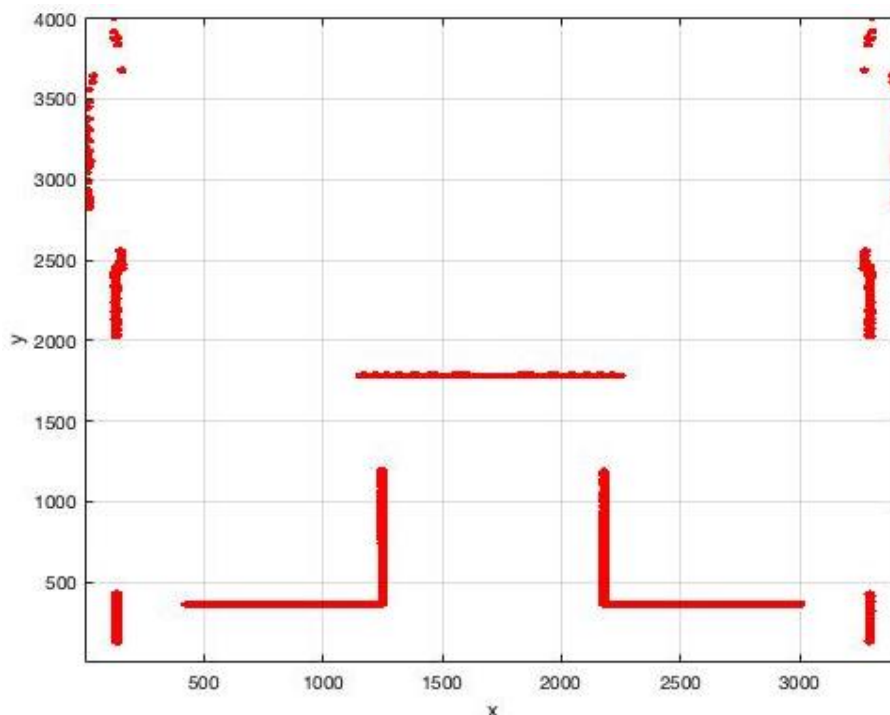


Fig. 5. Logical map supported by the Robotics Toolbox

Once input maps for the corresponding toolboxes were created, we can move to the navigation part, which is considered in the next section. An important observation here should be given to the trajectory points generated by those toolboxes. Namely, in order to control the mobile robot inside the proposed virtual environment and navigate around obstacles (see Fig. 1) the generated trajectory points must be converted back to the range of coordinates similar to the initial plot (see Fig. 2).

Experiment

The main goal of the experiment was to investigate the compatibility of our simulator with the Robotics System Toolbox and Robotics Toolbox. It was decided to demonstrate this compatibility by means of practical tasks, namely by navigation of the mobile robot inside the indoor environment. Additionally, the comparison analysis among these two toolboxes is provided. Before conducting experiments, the vision system must be calibrated. The calibration process and system model were described in the paper [11]. Since the calibration procedure is not the main goal of this paper, for the simplicity of computation and the experiment itself we assume that the parameters of the extrinsic calibration are known, namely the laser plane is parallel to the floor and the distance between a camera and a laser plane is known. These assumptions can be guaranteed by virtual environment. More details about the experiment are provided below. The OCamCalib toolbox [14] was used for obtaining intrinsic parameters of the fisheye camera.

Experiment Setup. In order to perform navigation of the mobile robot, several obstacles were placed inside the indoor environment (see Fig. 1). The initial position of the robot is (0; 0) meters, target position is (-0.2; 2.5) meters (see Fig. 2). Robotics System Toolbox uses a binary matrix of obstacles, whereas Robotics Toolbox uses a two-dimensional matrix. Consequently, the extracted laser beam belonging to obstacles was previously converted to the formats readable by these toolboxes. The next path between start and target points was created with corresponding toolboxes. After that, these trajectory points were converted back and transmitted via TCP/IP to the simulator for controlling the mobile robot. Finally, the experimental target position was compared with the position used for the experiment setup.

Results. Visual results of the conducted experiment are presented in Fig. 6 and Fig. 7 for the Robotics System Toolbox and for the Robotics Toolbox respectively. From these figures it can be seen that the mobile robot reached the desired target position without any visual misalignments. A more accurate comparison analysis is presented in Table 2. From Table 2 it can be seen that the coordinates of the experiment are similar to the coordinates from the experiment setup, the difference between them is very small, in the range of a few millimeters. Parameters of Table 2 also demonstrate that with the Robotics System Toolbox a mobile robot reached the target position faster than in case with the Robotics Toolboxes.

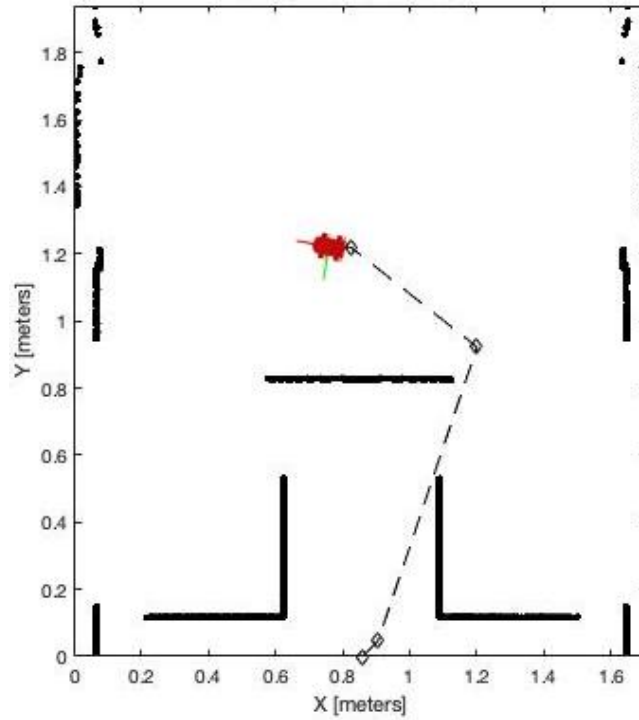


Fig. 6. Navigation to the target with Robotics System Toolbox

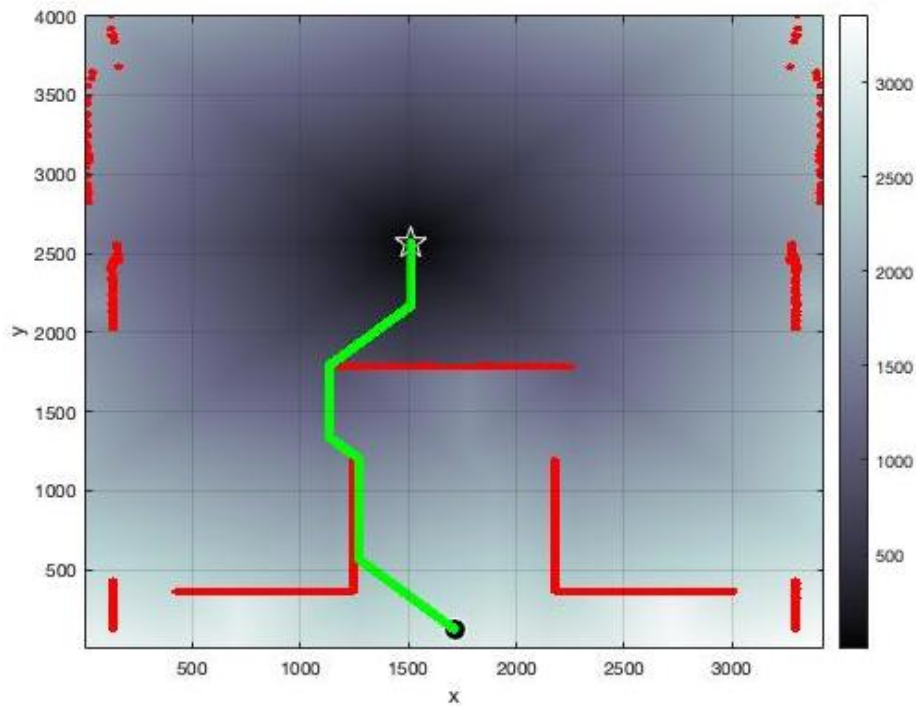


Fig. 7. Navigation to the target with Robotics Toolbox

Table 2

Comparison analysis

Parameter	Robotics System Toolbox	Robotics Toolbox
Input target, meters	(-0.200; 2.500)	
Experiment target, meters	(0.195; 2.505)	(0.199; 2.499)
Absolute error, meters	(0.005; 0.005)	(0.001; 0.001)
Operating time, seconds	67.68	178.85

Discussion. In general, it can be mentioned that Robotics System Toolbox provides a more powerful functionality for modeling and controlling mobile robots. For example, for navigation with Robotics System Toolbox it was possible to take into consideration the dimension of the robot in order not to collide with any obstacles. More details about the experiment results and merits of those toolboxes are discussed below.

Operating time: the experiment results showed that both toolboxes can generate and provide an accurate trajectory path to the target point. Meanwhile, for the Robotics System Toolbox this target point was reached almost three times faster in comparison with the Robotics Toolbox. The reason for this may be related to the format of the navigation map. The Robotics System Toolbox uses a binary matrix which occupies less space in memory and consequently processed faster than an ordinary two-dimensional matrix of the Robotics Toolbox.

Path modeling: in order to reach the target, the Robotics Toolbox moves the robot to the neighboring cell with the smallest distance to the target. The process is repeated until the robot reaches a cell with a distance value of zero which is the target. Meanwhile, the Robotics System Toolbox operates with a more advanced path planner which is based on the Probabilistic Roadmap (PRM). The PRM path planner constructs a roadmap in the free space of a given map using randomly sampled nodes in the free space and connecting them with each other.

Robot movements: instead of using discrete movements of the mobile robot provided by the Robotics Toolbox, the Robotics System Toolbox can provide more realistic navigation scenarios, namely with the aid of the Pure Pursuit controller. This controller is used in order to drive the simulated robot along the desired trajectory towards the target point.

Conclusion

The world quarantine has influenced the global economic system and resulted in the changes inside the educational system. New educational methods and approaches have become extremely important, as it allows teachers and students to overcome new challenges by maintaining the educational process. In this paper, we presented a publicly available extension for the Robotics System Toolbox and Robotics Toolbox, their principle features and capabilities. The software product presented in this paper showed the possibility of implementing virtual laboratories for studying mobile robots. This extension, represented in the form of the virtual laboratory has a number of important advantages for engineering education. We showed how the mobile robot equipped with the fisheye camera and structured light can be controlled by the Robotics Toolbox and Robotics System Toolbox. After that, we discussed and compared those toolboxes. The uniqueness of the extension also lies in its relative simplicity and low cost as the proposed extension does not require strong technical and economic resources, it can be run on basic operating systems. Educational systems which only require a computer can be named a useful teaching material. In our paper, we tried to demonstrate a case study of navigation. It is just a small part of what can be simulated or studied with virtual technologies. However, even by this small part we showed how the educational courses can be improved. We believe that improvements on the side of the educational process based on the virtual technologies will reflect the changes on the side of students, stimulating their interest, increasing motivation, and acquiring new practical skills.

References

1. Lenskiy A., Junho H., Dongyun K., Junsu P. Educational platform for learning programming via controlling mobile robots. In: *2014 International Conference on Data and Software Engineering (ICODSE)*. Bandung, Indonesia; 2014. P. 1–4. DOI: 10.1109/ICODSE.2014.7062695
2. Sadiku M., Adebo P., Musa S. Online Teaching and Learning. *International Journal of Advanced Research in Computer Science and Software Engineering*. 2018;8(2):73–75. DOI: 10.23956/ijarcsse.v8i2.549
3. Gomes A.S.A., Da Silva J.F., Teixeira L.R.D.L. Educational Robotics in Times of Pandemic: Challenges and Possibilities. In: *2020 Latin American Robotics Symposium (LARS), 2020 Brazilian Symposium on Robotics (SBR) and 2020 Workshop on Robotics in Education (WRE)*. Natal, Brazil; 2020. P. 1–5. DOI: 10.1109/LARS/SBR/WRE51543.2020.9307145
4. Ma L., Bai H., Dai Q., Wang H. Practice and Thinking of Online Teaching During Epidemic Period. In: *2020 15th International Conference on Computer Science & Education (ICCSE)*. Delft, Netherlands; 2020. P. 568–571. DOI: 10.1109/ICCSE49874.2020.9201803
5. Hu J., Zhang B. Application of Salesforce Platform in Online Teaching in Colleges and Universities under Epidemic Situation. In: *2020 International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*. Fuzhou, China; 2020. P. 276–279. DOI: 10.1109/ICBAIE49996.2020.00065
6. Kexin L., Yi Q., Xiaou S., Yan L. Future Education Trend Learned from the Covid-19 Pandemic: Take Artificial Intelligence Online Course As an Example. In: *2020 International Conference on Artificial Intelligence and Education (ICAIE)*. 2020. P. 108–111.
7. Bazarov S.E., Kholodilin I.Y., Nesterov A.S., Sokhina A.V. Applying Augmented Reality in practical classes for engineering students. *IOP Conference. Ser.: Earth and Environmental Science*. 2017;87:032004. DOI: 10.1088/1755-1315/87/3/032004

8. Robotics System Toolbox [Online]. Available at: <https://www.mathworks.com/help/robotics/> (accessed Feb. 6, 2021).
9. Corke P.I. A robotics toolbox for MATLAB. *IEEE Robotics & Automation Magazine*. 1996;3(1):24–32. DOI: 10.1109/100.486658
10. Afzal A., Katz D., Goues C., Timperley C. A Study on the Challenges of Using Robotics Simulators for Testing. Apr. 2020. [Online]. Available at: <https://arxiv.org/pdf/2004.07368.pdf> (accessed Feb. 8, 2021).
11. Kholodilin I., Li Y., Wang Q. Omnidirectional Vision System with Laser Illumination in a Flexible Configuration and Its Calibration by One Single Snapshot. *IEEE Transactions on Instrumentation and Measurement*. 2020;69(11):9105–9118. DOI: 10.1109/TIM.2020.2998598
12. Scaramuzza D., Martinelli A., Siegwart R. A Flexible Technique for Accurate Omnidirectional Camera Calibration and Structure from Motion. In: *Fourth IEEE International Conference on Computer Vision Systems (ICVS'06)*. New York, NY, USA; 2006. P. 45–45. DOI: 10.1109/ICVS.2006.3
13. Saxena S., Jain S., Tripathi S., Gupta K. Comparative Analysis of Image Segmentation Techniques. In: *Advances in Communication and Computational Technology*. Springer, Singapore; 2021. Vol. 668. P. 317–331. DOI: 10.1007/978-981-15-5341-7_26
14. Scaramuzza D. OCamCalib: Omnidirectional Camera Calibration Toolbox for MATLAB [Online]. Available at: <https://www.sites.google.com/site/scarobotix/ocamcalib-toolbox> (accessed Feb. 12, 2021).

Information about the authors

Ivan Yu. Kholodilin, Cand. Sci. (Eng.), Ass. Prof., Lecturer of the Department of Electric Drive, Mechatronics and Electromechanics, South Ural State University, Chelyabinsk, Russia; kholodilini@susu.ru.

Maksim A. Grigorev, Dr. Sci. (Eng.), Prof., Head of the Department of Electric Drive, Mechatronics and Electromechanics, South Ural State University, Chelyabinsk, Russia; grigorevma@susu.ru.

Victor A. Kushnarev, Master's Student of the Department of Electric Drive, Mechatronics and Electromechanics, South Ural State University, Chelyabinsk, Russia; a55538a@yandex.ru.

Nikita V. Savosteenko, Senior Lecturer of the Department of Electric Drive, Mechatronics and Electromechanics, South Ural State University, Chelyabinsk, Russia; savosteenkonv@susu.ru.

Danila V. Spitsin, Student, South Ural State University, Chelyabinsk, Russia; yanda.32@mail.ru.

Oleg I. Osipov, Dr. Sci. (Eng.), Prof., Prof. of the Department of Automated Electric Drive, National Research University "Moscow Power Engineering Institute", Moscow, Russia; osipovoi2011@yandex.ru.

Информация об авторах

Холодilin Иван Юрьевич, канд. техн. наук, доц., преподаватель кафедры электропривода, мехатроники и электромеханики, Южно-Уральский государственный университет, Челябинск, Россия; kholodilini@susu.ru.

Григорьев Максим Анатольевич, д-р техн. наук, проф., заведующий кафедрой электропривода, мехатроники и электромеханики, Южно-Уральский государственный университет, Челябинск, Россия; grigorevma@susu.ru.

Кушнарёв Виктор Александрович, магистрант кафедры электропривода, мехатроники и электромеханики, Южно-Уральский государственный университет, Челябинск, Россия; a55538a@yandex.ru.

Савостеенко Никита Вадимович, старший преподаватель кафедры электропривода, мехатроники и электромеханики, Южно-Уральский государственный университет, Челябинск, Россия; savosteenkonv@susu.ru.

Спицин Данила Валерьевич, студент, Южно-Уральский государственный университет, Челябинск, Россия; yanda.32@mail.ru.

Осипов Олег Иванович, д-р техн. наук, проф., проф. кафедры автоматизированного электропривода, Национальный исследовательский университет «МЭИ», Москва, Россия; osipovoi2011@yandex.ru.

The article was submitted 15.07.2024; approved after review 21.09.2024; accepted for publication 21.09.2024.

Статья поступила в редакцию 15.07.2024; одобрена после рецензирования 21.09.2024; принята к публикации 21.09.2024.