Engineering Educators Bringing the World Together
2025 ASEE Annual Conference & Exposition
Palais des congrès de Montréal, Montréal, QC • June 22–25, 2025
Paper ID #47478

Vertical lab Integration with myCobot Robotic Arms

Dr. Nebojsa I Jaksic P.E., Colorado State University Pueblo

NEBOJSA I. JAKSIC earned the Dipl. Ing. (M.S.) degree in electrical engineering from Belgrade University (1984), the M.S. in electrical engineering (1988), the M.S. in industrial engineering (1992), and the Ph.D. in industrial engineering from The Ohio State University (2000). Currently, he is a Professor at Colorado State University Pueblo. Dr. Jaksic has over 100 publications and holds two patents. His interests include robotics, automation, and nanotechnology. He is a licensed PE in the State of Colorado, a member of ASEE, and a senior member of IEEE and SME.

Vertical Lab Integration using myCobot Robotic Arms

Abstract

This work describes a framework consisting of five vertically integrated robotic lab modules developed for two mechatronics engineering programs (BS and MS) based on small myCobot cobots produced by Elephant Robotics. The required courses using myCobot robotic tasks are the first-year course EN 101 Introduction to Engineering, the third-year course EN 362L Introduction to Mechatronics Lab, the fourth-year course EN462L Industrial Robotics Lab, the fourth/fifth-year course EN563 Intelligent Robotics, and the fourth/fifth-year course EN 513 Artificial Intelligence. These courses span our BS and MS programs in mechatronics engineering. With progressively more advanced robotic tasks, students increase their robotic programming skills, starting with myBlockly graphical programming environment, progressing through Python3 textual language, learning Robotic Operating System (ROS), and finally using vision and AI libraries, all employing the same small inexpensive manipulators. As expected, qualitative student responses addressing the above robotic labs show labs' positive impact on students' learning and attitudes. Also, the intention of this work was to address various students' robotic learning needs through a discussion of various myCobot computer hardware and software implementations.

Introduction

Engineering education literature strongly suggests that robotic labs attract students to engineering while multiple labs with the same mechanical robotic hardware create scaffolding effect which improves learning. The main goal of this work is to provide robotics professors with an affordable set of cobot laboratory tools. The work presents a set of vertically integrated robotic labs based on an inexpensive robotic platform to increase students' robotic programming skills and understanding of robots and cobots through scaffolding and progressively more advanced lab exercises.

What follows are sections on previous work emphasizing experiential learning, then curricular context, metrics and results, and finally a summary and future work.

Previous Work

Practical laboratory experiences including engineering labs and projects represent essential elements of learning [1], [2]. As part of intensive laboratory experiences, robots have had a longstanding positive impact on education of students at all levels. Small, wheeled, programable mobile robots like LEGO Mindstorm series have been used as motivational tools to attract students to STEM fields in general [3], as well as to help students (and teachers) learn how to program [4] - [6]. However, at the practical level of industrial robot programming, the use of industrial manipulators for teaching programming robotic tasks was often the only option. Expensive hardware, proprietary software, and required safety measures made programming of industrial manipulators difficult. A lightweight, joint torque-controlled robot by KUKA was designed to operate alongside humans [7]. However, the high robot price (over $100,000) made it inaccessible for robotics education. In 2017, a $10,000, lightweight, tactile robotic manipulator by Agile Robots, Franka Emika, was introduced for robotics research in AI and machine learning, as well as for robotic education [8]. The above two devices belong to the collaborative robots or cobots family of robots since they are designed to work alongside humans in shared workspace by being lightweight and being able to detect/avoid obstacles. Wayne State University and Oakland Community College developed teaching modules specifically for learning cobots [9]. In 2021, a lightweight (less than 1 kg), inexpensive (about $1,000), myCobot 280 Jetson Nano by Elephant Robotics was introduced. While there are some limited applications in robotics research [10], there were no publications in engineering education literature implementing myCobots.

Curricular Context

Colorado State University Pueblo (CSUP) is an accredited, regional comprehensive, Hispanic-serving institution (HSI) enrolling about 3500 students. The university is a part of the Colorado State University System. The School of Engineering, a part of the College of Science, Technology, Engineering, and Mathematics (CSTEM) hosts five undergraduate BS programs: Mechatronics Engineering (BSME), Industrial Engineering (BSIE), Civil Engineering (BSCE), Civil Engineering Technology (BSCET), and Construction Management (BSCM), as well as two MS programs, one in Mechatronics Engineering (MSME) and one in Industrial and Systems Engineering (MSISE). It is worth mentioning a 3+2 program offered at CSUP where, after completion, students receive simultaneously BS and MS degrees in ME. The series of labs described here includes mostly students from BSME and MSME programs.

Characteristics of MyCobot Robotic Arms

While Elephant Robotics manufactures a variety of robots, this work focuses on their cobot myCobot 280 series, specifically the version using NVIDIA Jetson Nano AI System on a Module (SoM). MyCobot 280 arms are small (work envelope radius of 280 mm), light (850 g without a gripper), inexpensive (about $1,000), six degree of freedom (RRRRRR), electrically powered (60W) robotic manipulators that can carry a 250 g payload. The robots can be equipped with a camera and an electric gripper or a suction cup. This robotic arm configuration is standard across the series. The flexibility of myCobot robots stems from the variety of the controllers used (Arduino, M5stack, Raspberry Pi, or Jetson Nano) with addition of an auxiliary controller (Atom M5), as well as the software running on various operating systems (Ubuntu, Windows, and macOS), programming environments like Robot Operating System (ROS 2) shell or Visual Studio Code, computer languages like myBlockly, Python, or JavaScript, and libraries like myCobot for Python, OpenCV, and TensorFlow. This allows great flexibility in customization for different robotic labs suited to different students' educational levels and programming language choices.

Elephant Robotics has created a number of myCobot robotic arms. Fig. 1. shows a myCobot 280 arm with a camera and mounting hardware. The robot must be attached to the table, otherwise it would topple over if the action commanded moves the robot's center of gravity outside its base, which is easily achievable since the arm is light. In addition, the latest Ubuntu distribution with corresponding ROS 1 and ROS 2 shells, myStudio, myBlockly, and other software corresponding to the computer hardware used are preloaded. Connecting a keyboard, a mouse, and a monitor is all that is needed to start. When the Jetson Nano finishes bringing up the system, one should run myStudio to upgrade the firmware.

Fig. 1. Elephant Robotics myCobot 280 Jetson Nano Robotic Arm

MyBlockly is a graphical programming language similar to Scratch. MyBlockly window, shown in Fig. 2, has two tabs, Blockly and Python. When clicked, the Python tab shows the Python version of myBlockly commands as depicted in Fig. 3.

Robotic Laboratory Modules

The required BSME and MSME courses implementing myCobot 280 Jetson Nano cobots to perform a set of robotic tasks are the first-year course EN 101 Introduction to Engineering, the third-year course EN 362L Introduction to Mechatronics Lab, the fourth-year course EN462L Industrial Robotics Lab, the fourth/fifth-year course EN563 Intelligent Robotics, and the fourth/fifth-year course EN 513 Artificial Intelligence. Each module consists of a robotic task to be performed and a report. Five robotic tasks corresponding to each of the five courses are listed in Table 1 and further described in this section. Table 1 also includes a mapping of the robotic lab student learning outcomes (SLOs) to ABET EAC Criterion 3: student outcomes. The lab reports consist of four sections answering the following questions: What problems did you encounter? How did you solve them (include your solution/program)? What did you learn from this robotic task? What did you like the most about this robotic task?

The base line comparison of myCobots ($1000) is with LEGO Education SPIKE Prime ($400) sets used previously in the Introduction to Engineering course. They both use Scratch-based drag-and-drop languages. The SPIKE environment is suitable for development of programs dealing with small mobile vehicles, whereas myBlockly environment is created for solving industrial manipulator tasks. MyBlockly includes commands for arm movements in both joint and Cartesian coordinates as well as opening/closing of two types of grippers. Also, both robotic development environments can use Python 3, myCobots in its standard form and LEGO Education SPIKE Prime as MicroPython (Python 3 for microcontrollers). To summarize, if there is a need to teach some basic programming with a Scratch-based programming language (or Python) then the SPIKE Prime is a less expensive and viable choice. However, myBlockly and Python 3 with included myCobot libraries address a robotic knowledge base which is in line with current industrial manipulators. LEGO Education Spike Prime set is not suited for programming of industrial manipulators.

For subsequent labs, myCobots robotic arms at about $1,000 with grippers are compared to small industrial manipulators ABB IRB 120 at $15,000 without grippers. While the two have the same number of degrees of freedom (6 DOF), there is a huge difference in load capabilities of the two robots, 4 kg for IRB 120s and 250 g for myCobots, as well as weight, 25 kg for IRB 120s and 1 kg for myCobots.

IRB 120s are not designed to work alongside people while collaborative robots (cobots) like myCobot are. For safety reasons, IRB 120s are either fenced in or there are clear safety designations marking the safety zones. MyCobots are much lighter, safer, and easier to store when not in use. This feature makes myCobots easy to transport and setup for in-lecture robotic demonstrations. However, IRB 120s are much sturdier (mostly metal construction) than myCobots.

Finally, IRB 120s use a proprietary development environment RobotStudio and a proprietary robotic language RAPID which only work with ABB robots. In contrast, MyCobots use well known, ubiquitous, and free software like Ubuntu, Python 3, ROS 2, OpenCV, etc.

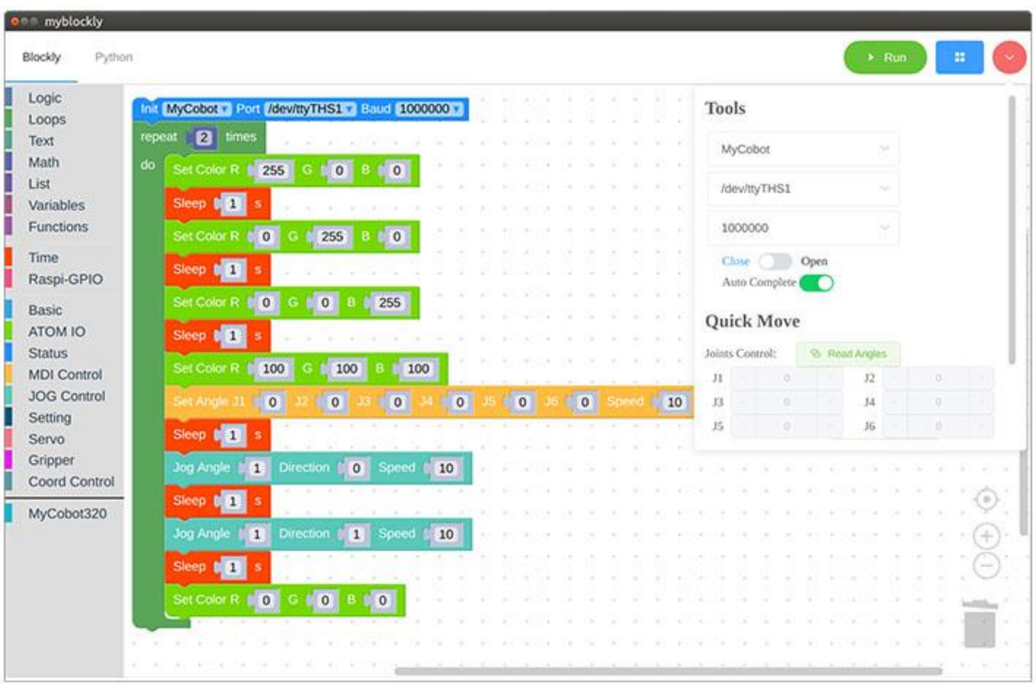

Fig. 2. MyBlockly window

Fig. 3. Python translation of myBlockly code

Table 1. MyCobots robotic labs, courses, and SLOs

Cobot Task	Course	Year	SLOs	ABET EAC Criterion 3
Programming Atom 5 LED matrix and simple arm motions using MyBlockly	Introduction to Engineering	One	1. An ability to program simple robotic tasks in MyBlockly and ATOM 5 2. An ability to program physical industrial type cobots vs. programming LEGO SPIKE mobile robots	1, 3, and 6
Simple robotic task using an adaptive gripper and programming in Python	Introduction to Mechatronics Lab	Three	An ability to use adaptive grippers and Python in programming simple robotic tasks	1, 2, and 6
Implementation of a pick-and-place robotic task in Python	Industrial Robotics Lab	Four	An ability to program pick-and-place robotic tasks using Python	1, 2, and 6
Using ROS 2 and a camera to program complex robotic tasks	Intelligent Robotics	Four/ Five	An ability to use ROS 2 with camera(s) in programming complex robotic tasks	2, 6, and 7
Implementation of object recognition and stacking using AI methods	Artificial Intelligence	Four/ Five	An ability to use cameras with OpenCV (or TensorFlow) AI libraries for solving complicated robotic tasks	2, 6, and 7

Robotic task one: programming an Atom 5 LED matrix and simple arm motions using MyBlockly

EN101 Introduction to Engineering, a two-credit-hour course, is structured to allow each professor in the program to meet with first-year undergraduate students and demonstrate some interesting aspects of the courses they teach. While we are using a Jetson Nano-based myCobot, an Arduino-based robot is sufficient for a short robotic demonstration for this course. In class, robot programming can be quickly demonstrated using myBlockly drag-and-drop graphical programming language.

The first robotic task consists of the following subtasks.

1. Verify that the robot is operational by remotely controlling the robot by using your phone. You will need to download the Mycobot-Controller app to your phone.
2. Using myBlockly, program the robot to perform the following actions twice.
    a. Turn the LED matrix of the Atom 5 red
    b. Hold for one second
    c. Turn the LED matrix green
    d. Hold for one second
    e. Turn the LED matrix blue
    f. Hold for one second
    g. Turn the LED matrix to color 100, 100, 100
    h. Move all joints to joint angles of 0 at the speed of 10
    i. Hold for one second
    j. Jog continuously joint one in the clockwise direction (decreasing angle) at the speed of 10
    k. Hold for one second
    l. Jog continuously joint one in the counterclockwise direction (increasing angle) at the speed of 10
    m. Hold for one second
    n. Set the LED matrix color to black (0, 0, 0)

Fig. 2 shows a MyBlockly program that performs the above task, while Fig. 3 shows the corresponding Python code for the same task.

Robotic task two: using an adaptive gripper and programming myCobot in Python

EN362L Introduction to Mechatronics Lab is a one credit required course that introduces labs that build students' programming skills with microcontrollers. Again, even though the Arduino-based myCobot robots are sufficient for this application we use Jetson Nano based myCobots. Before starting the robotic task, students are exposed to Linux OS and basic Python3 commands. They also learn some of the Python integrated development environments (IDEs) like IDLE, PyCharm, and Visual Studio Code (VS Code).

The second robotic task is implemented in two sessions (two hours each) of the lab. Students are required to use Python 3 and an IDE of their own choice with pymycolab Python library for myCobot 280. For this task an adaptive gripper is attached to the robot. The robotic task consists of the following subtasks.

1. Perform testing of the Atom 5's LED matrix by flashing red, green, and blue in intervals of 2 seconds for each color. When done, set the LED matrix to green
2. Bow the robot three times towards the programmers
3. Open and close the gripper two times

Robotic task three: implementation of a pick-and-place robotic task in Python

EN 462L Industrial Robotics Lab is a one-credit course that introduces typical robotic tasks like material handling, packing, palletization, welding, painting, etc. A more powerful, Raspberry Pi based myCobot running Python on Ubuntu can be used. However, not to duplicate resources, a myCobot 280 Jetson Nano is used in this course as well. The robot is equipped with an adaptive gripper. After students program an ABB IRB 120 industrial robot using RobotStudio and RAPID, students program a pick-and-place robotic task in Python using myCobot 280.

The third robotic task is a pick-and-place task. In general, three 2 cm cubes (blue, red, and green) are moved from their original location and stacked on top of a cylinder. Then, the cubes are placed back but in a different order. The position of each cube is known.

Robotic task: using Python, program a myCobot 280 Jetson Nano robot equipped with an adaptive gripper to stack the three cubes (2 cm) on top of the disk (5 cm radius, 2 cm height) in order (blue, red, and green) from top to bottom and then redistribute them back in order red, green, and blue (from left to right with respect to the programmer). After stacking the cubes, the robot should pause for 2 seconds. Also, when done, the robot should return home (in joint coordinates it is 0, 0, 0, 0, 0, 0). The task starting position is with the robot in its home position. Fig. 4. shows stacked cubes. An alternative version of the task, where students stack three cylinders on top of each other, is shown in Fig. 5.

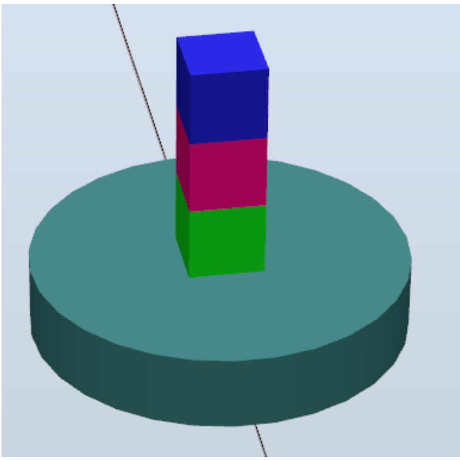

Fig. 4. Stacked cubes on top of a cylinder

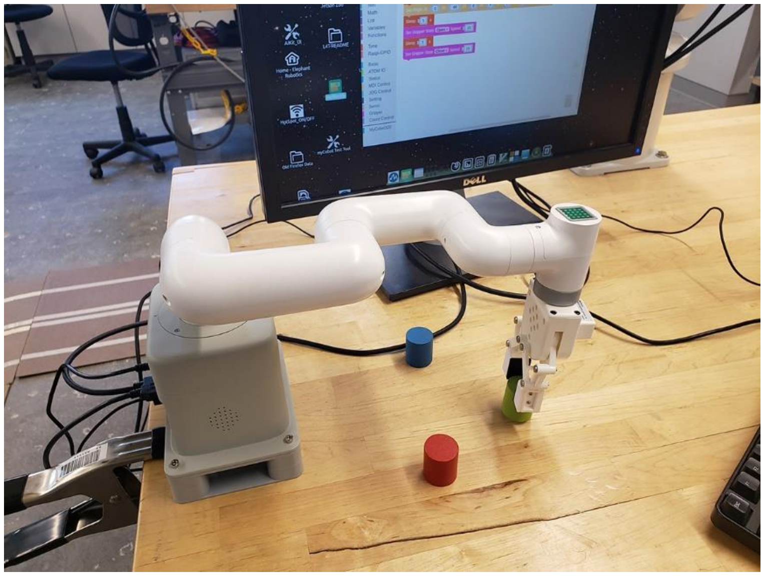

Fig. 5. MyCobot stacking three cylinders

Robotic task four: Using ROS 2 and a camera to program complex robotic tasks

EN563 Intelligent Robotics is a 3-credit course offered to the senior-standing undergraduate (elective) and the first-year MS Mechatronics Engineering students (required). While the course does not have a dedicated lab session, hands-on lab experiences are provided through homework assignments. It is estimated that students spend 10 to 15 hours on this assignment. Students use myCobot 280 Jetson Nano robot to introduce ROS 2 shell. This lab can still be accomplished with myCobot Pi. Students learn how to program more complicated robotic tasks like sorting. They also learn how to use cameras. For this task, a myCobot 280 is equipped with an adaptive electric gripper and a camera. Students repeat the third robotic task using ROS 2 shell. They also use cameras (one per robot) to monitor the progress of the task.

Robotic task five: implementation of object recognition and stacking using AI methods

EN513 Artificial Intelligence is a 3-credit required course offered to graduate students. Undergraduate students of senior standing may enroll, but they take this course as an elective. In the lab, students use myCobot 280 Jetson Nano with OpenCV or TensorFlow for solving image recognition and classification problems as well as simultaneous localization and mapping (SLAM) problems. For this task, the robot is equipped with a gripper and a camera.

The fifth robotic task is a repeat of the previous task with one significant difference; the cubes are placed randomly on the three positional markers. An additional challenge is offered by placing the three cubes anywhere in the work envelope of the robot.

Metrics and Results

While this work describes a framework of five robotic tasks vertically integrated through five years, it mainly deals with a new and affordable robotic platform (myCobot) and its impact on students' learning robotics. The assessment metrics for the above set of robotic labs are improvements in programming skills, increase in knowledge of robotics, and increases in affinity towards robotics and programming. Data for qualitative analyses of these variables are obtained from student lab reports and individual interviews with students. The Encountered Problems and Solutions section of lab reports provides an insight into students' proficiency improvements in programming robotic tasks as well as programming in general. These results were all positive. Self-reflections through answering the question "What have I learned from this lab experience?" provide further insight into students' gain in knowledge and confidence in their programming skills. Finally, the question "What I liked the most about this lab experience?" forces students to think positively about their experiences and thus learn better. The retention rates for the first-year students did not change when the LEGO SPIKE robotic lab was replaced by the myCobots robotic lab.

Students' testimonials show their pride when they were able to program lab robots to accomplish the robotic tasks as specified. Statements like *"Finally some real robots, not LEGO toys!"* and *"I joined this program to learn about robots, and here they are. Thank you!"* as well as *"While I was afraid of big industrial robots, I am very comfortable with cobots now."* were not uncommon.

Since most of the engineering students have not been introduced to Linux operating system before, some students found it difficult to work in Ubuntu. Most of the students claimed that they learned much. Finally, all students liked (or loved) the robotic programming experiences, which is understandable because they are all enrolled in one of the mechatronics engineering programs.

It is worth mentioning that some students complained about loud high-pitch noise that myCobots produced.

Summary and Future Work

In this work, a set of related robotics labs using cobots to create a vertically integrated framework for mechatronics engineering students is described. The framework, consisting of five vertically integrated robotic lab modules, is developed for two mechatronics engineering programs (BS and MS) based on small myCobot robotic manipulators. Students in the first, third, fourth, and fifth year were positively affected. For robotics faculty, the low cost and the ease of use of these robotic manipulators present a lucrative alternative to other robotic platforms like LEGO SPIKE sets. The state-of-the-art software (Ubuntu, Python, ROS 2, etc.) enables teaching advanced computing concepts and their applications in robotics. With progressively more advanced robotic tasks, students increase their robotic programming skills and understanding of robotics, all using the same small inexpensive cobots. As expected, qualitative student responses show positive impact on students' learning, programming skills development and attitudes towards robotics. Also, this work addressed various students' robotic learning needs through a discussion of various myCobot computer hardware options.

In the future, it is planned to extend the robotic tasks to obstacle detection and avoidance as well as digital twins. Also, the use of cobots in education will be emphasized because of cobots safety features.

Bibliography

[1] J. Dewey, *Experience and Education*, Macmillan, N.Y., 1939.

[2] D. A. Kolb, *Experiential Learning: Experience as the Source of Learning and Development*, Prentice Hall, Englewood Cliffs, N.J., 1984.

[3] C. Pomalaza-Raez and B. H. Groff, "Retention 101: Where Robots Go … Students Follow," *Journal of Engineering Education*, vol. 92, No. 1, pp.85-90, January 2003. https://doi.org/10.1002/j.2168-9830.2003.tb00742.x

[4] D. Alimisis, "Robotics in Education & Education in Robotics: Shifting Focus from Technology to Pedagogy," in David Obdrzálek (ed.) *Proceedings of the 3rd International Conference on Robotics in Education*, pp. 7-14, September 13 – 15, 2012.

[5] N. I. Jaksic and D. E. Spencer, "Multidisciplinary Robotics Experiment: LEGO Mindstorms NXT Bluetooth Convoy," *International Journal of Modern Engineering*, Vol. 10, No. 1, pp. 5 – 10, Fall/Winter 2009.

[6] D. E. Spencer and N. I. Jaksic, "A Multidisciplinary Robotics Learning Environment: What Mindstorms and DARPA Urban Challenge Have in Common," *Computers in Education Journal*, *Special issue on Novel Approaches to Robotics Education*, Vol. 1, No. 3, (July – Sept. 2010), pp. 32 – 40, 2010. https://coed.asee.org/2010/07/03/a-multidisciplinary-robotics-learning-environment-what-mindstorms-and-darpa-urban-challenge-have-in-common/

[7] R. Bischoff, et al.," The KUKA-DLR lightweight robot arm - A new reference platform for robotics research and manufacturing," in Proc. 41st Int. Symp. Robotics (ISR)/6th German Conf. Robot. (ROBOTIK), pp. 741–748, Jun. 2010.

[8] S. Haddadin, "The Franka Emika Robot – A Standard Platform in Robotics Research," *IEEE Robotics and Automation Magazine*, Vol. 31, No. 4, pp. 136 – 148, December 2024. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10693652

[9] A. Djuric, J. Rickli, J. Sefcovic, D. Hutchison, and M. M. Goldin, "Integrating Collaborative Robots in Engineering and Engineering Technology Programs," ASME 2018 International Mechanical Engineering Congress and Exposition, Pittsburgh, PA, Published on Jan. 15, 2019. https://doi.org/10.1115/IMECE2018-88147

[10] K. Kawaharazuka, N. Kanazawa, K. Okada, and M. Inaba, "Self-Supervised Learning of Visual Servoing for Low-Rigidity Robots Considering Temporal Body Changes," *IEEE Robotics and Automation Letters*, Vol. 7, No. 3, pp. 7881 - 7887, July 2022.