IOP Conference Series: Materials Science and Engineering

CVCER: Robot to Learn Basics of Computer Vision and Cryptography

To cite this article: A Yeleussinov *et al* 2018 *IOP Conf. Ser.: Mater. Sci. Eng.* **417** 012013

IOP Conf. Series: Materials Science and Engineering **417** (2018) 012013 doi:10.1088/1757-899X/417/1/012013

CVCER: Robot to Learn Basics of Computer Vision and Cryptography

A Yeleussinov[1,2], T Islamgozhayev[2], M Satymbekov[2] and A Kozhagul[3]

[1]al-Farabi Kazakh National University, 71 al-Farabi Avenue, Almaty, Kazakhstan
[2]Institute of Information and Computational Technologies, 125 Pushkin str. Almaty, Kazakhstan
[3]Institute of Mathematics, Physics and Informatics, Kazakh National University named after Abai, Dostyk ave. 13, Almaty, Kazakhstan

arman.04081991@gmail.com

Abstract. This paper describes the design of CVCER (Computer Vision and Cryptography Educational Robot) - the working model of the mobile robot used as a training stand for students. We describe the hardware and software of the robot. Hardware part of the robot can be used to teach the basics of robot architecture and software part can be used to teach the basics of computer vision and cryptography. The main goal of the project was to create a low cost, re-configurable and re-programmable robotic platform that could be used to teach the basics of the subjects described above and to test the algorithms written by students having internships in our institute and partner university.

1. Introduction

Research and development in the area of robotics during past decades brought new subjects into an educational program of schools and universities. Highly motivated scientists and hobbyists are creating and developing new platforms, educational programs, and methods of teaching STEM subjects like physics, mathematics, engineering, etc [1-3]. However, most of robots are vendor-dependent, without source code and expensive. Aim of this paper is to describe the process of continual development of a robotic platform, which can be used in teaching STEM subjects and basics of Robotics, Control Theory, Machine Vision, Information Security, and building blocks of which can be bought or 3D printed and used to test robotics projects or in competitions. To reach this aims we developed CVCER - cheap, flexible and multi-functional robotic platform. CVCER's hardware part is built using Arduino platform, which is most popular hardware-platform that gives open architecture and lot of compatible sensor sets.

CVCER is an educational robotic platform (Figure 1) that was firstly developed as a platform for practical realization of Machine Vision algorithms [4], now the system evolved and we wanted to be able to teach Information Security basics with the help of the system. From this field we wanted to teach the basics of information retrieval between the operator and robotics system, that is why to ensure the privacy of sent commands we developed a system with an information security using and NPNs algorithm described in [5]. In our work the main aim is not to teach the basics of constructing physical robot by joining different 3D printed objects but to foster the teaching of development of control, pattern recognition, and information security algorithms and immediate testing of results on the platform because the real-life tests drive more attention in training process than simulation or

IOP Conf. Series: Materials Science and Engineering **417** (2018) 012013 doi:10.1088/1757-899X/417/1/012013

theoretical results [6-8]. To make the process easier, OpenCV library was used for the purpose of implementation of computer vision teaching software.

2. Platform

CVCER body is made from 3D printed chassis with modification - all wheels were replaced by omni-wheels that adds flexibility [4]. Body frame is made from perforated aluminum that makes platform flexible to any modification and attachment of additional equipment.

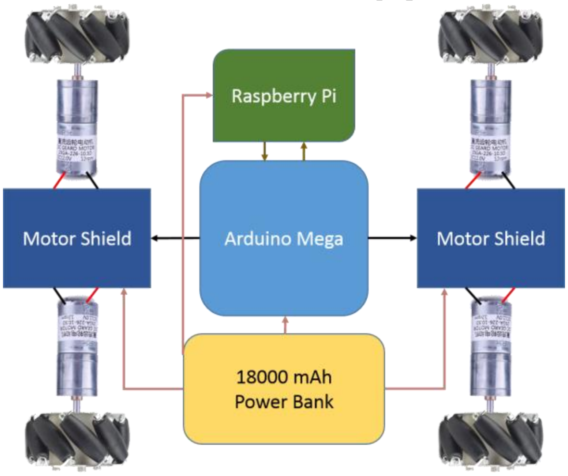

Figure 1. Platform Architecture

For now, 18000 mAh power bank with 5-20V outputs gives the system 80-100 minutes of life to power the platform. Parts can be replaced and new parts can be easily adjusted to the base that has drilled holes.
Movement of the platform is done using four motors: two motors on each side of the platform. Each motor can be programmed to move on its own, which gives more flexibility in motion planning tasks, because you can control almost any behavior of the platform by using the torque from each of them.

Figure 2. CVCER robotic platform

Raspberry Pi 3 Model B and Arduino Mega 2560 are selected as the brain and motion control unit of the robot. Raspberry Pi includes 802.11n Wi-Fi, Bluetooth 4, and Quad Core 1.2GHz Broadcom BCM2837 64bit CPU with 1Gb RAM [9]. Such characteristics and capability of using Linux-family operating system makes it one of the most commonly used single-board computers for robotics projects. Moreover, it makes possible the usage of OpenCV and programmable GPIO's - functions that perfectly suit the aim of our project. Because we have to run object recognition and data decryption programs, as well as sending control commands to Arduino to perform motion.
Arduino is very popular platform for robotics. Its flexibility and availability makes it ideal for fast prototyping and experiments. Arduino Mega 2560 is used as the main controller. This board is based

on the ATmega2560 chip and has 54 digital input/output pins, 16 analog pins [10]. Description of mechanical and electronic parts you can read from our previous work [4].
Platform makes use of mecanum wheels to add the flexibility and fast side-to-side movements [11,12]. It can surge (forward/backward), sway (left/right) without turning to sides, and yaw around the Z-axis like crawler which gives it two more controllable degrees-of-freedom, thus, making the system holonomic.
In the given system, four wheels are placed like standard wheels of four-wheeled car, it gives the system stability and simplifies control approach [13].
Motion control is written using IICTv3.h library used in previous version of the system. Additional functionality was added including the motion control of the 2DOF camera mount. Camera mount is a pan and tilt system driven by two Hitec HS-785HB servos capable of working with maximum payload of 2.5 kgs.
Tests were done by using the combination of two Arecont Vision cameras with optical zoom lenses (this approach requires router and an additional source of power to be placed, because AV cameras are network cameras which consume a lot of energy) and paired webcams, availability of USB ports on Raspberry PI makes replacement by any affordable USB webcams possible.
We estimate the cost of the entire system to be around 500 US dollars (with simple 5MP webcam pairs).

3. Software
As mentioned earlier, the robot is composed of several parts. Therefore, the latest complete software consists of a combination of machine vision, data decryption, and motion control blocks (Fig. 3 software architecture). We believe that last year students are familiar with the basics of programming languages.

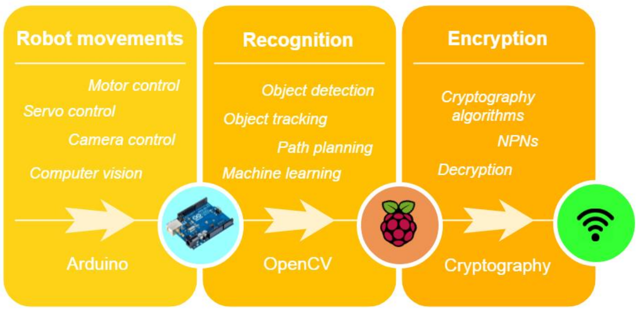

Figure 3. Architecture of the software

Students can use the programming language based on their choice, programming environment and methods, different algorithms, and, if necessary, additional devices (motors, sensors, etc.). But the introductory software is written as the console application in C++ language using OpenCV library, and several System classes, our main aim was not to build "put and go" system, but to build a system that can help to teach full stack of the development process - from combining the hardware part to programming the basic behaviors, to learning the object detection, tracking, and at the end, to learn to encrypt and decrypt incoming and outgoing messages. Figure 4 illustrates packages included into CVCER library.

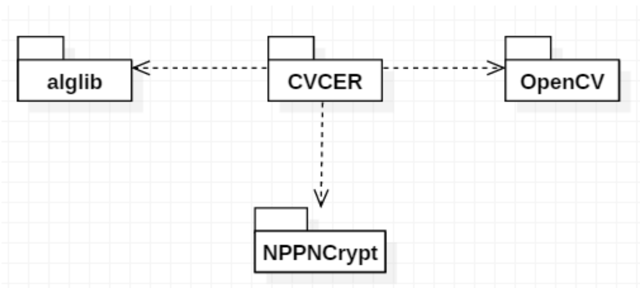

Figure 4. Packages diagram

3.1. Programming basic behaviours of the robot

At this stage, user needs to move actuators connected to the Arduino platform through motor drivers (originally 4 motors and 2 servomotors). Code should be written in C/C++ language. For this purpose, user can program on his own or call our IICTv3.h library key functions for motion programming (like forward(seconds), backwards(seconds), left(seconds/angle), right(seconds/angle), etc). The main purpose of this stage is to carry out all possible movements of the robot and complete the motion control task. Main library (CVCER) includes two modules - IICTv3 (responsible for motion control), NPPNCrypt (encryption/decryption). OpenCV and alglib libraries are ready to use simply by "including" them.

3.2. Object recognition algorithms and the robot movement based on machine vision

Since the platform is designed to be used at different levels of learning, the connection of robot motion and computer vision parts should be made to meet this differentiation (it should be made to make everyone familiar with the connection of these two parts). A combination of two cameras makes it possible to use stereo vision approaches and camera mount with 2 DOF allows user to learn basics of trajectory generation, motion planning and perform object tracking and mecanum wheels makes the object following simpler by making it possible without turning the whole platform. OpenCV's basic template matching methods (square difference matching, correlation matching, correlation coefficient matching, normalized square difference matching, normalized correlation matching, and normalized correlation coefficient matching) were used to perform the task of recognition of the object, the corresponding functions were written to be able to test each method by using just single line of code. Object tracking and following tasks were made to give simple two variable outputs that denotes the distance and angle to object from the optical center of the cameras.

3.3. Application of the encryption algorithm for robot control

There is a need to encrypt the security of data exchange between the robot and other robots (in multiagent systems) or the operator. To that end, the user can use the encryption algorithm based on non-positional polynomial notations developed by our colleagues [5]. Keys of different lengths are required for encryption and decryption between the operator and the robot. It is generated in a special program on the personal computer used by the operator and is sent to the computer on a mobile robot through WiFi. Of course, sending simple control commands (forward/backward, left/right, etc) is not the true usage of the algorithm, because messages sent could be easily caught in the middle and copied by intruders, the main idea of this part was to encrypt combination of commands or complex tasks.

4. Conclusion

The article gives brief description of the motion control, computer vision strategies and encryption algorithm used in the development of mobile educational robot. It describes the achieved results, ideas used and in future report will be described the ready-to-use software package with friendly GUI, rebrained and finished backend and polished body. As the platform has the ability to hold different sensors and additional hardware, by adding them and including additional functions in combination with our library users can modify and enhance the capabilities of the system. Presented library includes commands for setting up pins, driving motors, turning the platform, pre-written commands for combination of joystick buttons (if user is going to use them), basic computer vision algorithms and encryption algorithm. The mobile platform was developed to help beginners – enthusiasts, school and university students – to become familiar with the basics of development of robot systems, which involves different fields of study. We picked up trending areas of robotics and computer science fields and tried to make the robotics software development and motion programming easier, more understandable by providing live examples of how things made and what processes robot developers are undergoing during the development cycle. We also used most popular computer vision library, encryption algorithm and motion control library developed in our institution to make it easy to get the documentation (in case of OpenCV) and help from community as well as to describe the technologies from first person view (as we have the direct connections with each of the developers of the given system). Future report will also be about the development of the described system, but more detailed insights will be given to each part.

References

[1] Bellas F. et al. *The Robobo Project: Bringing Educational Robotics Closer to Real-World Applications.* J. Adv. Intell. Syst. 630 (2018), 226-237.
[2] W. Lepuschitz et al. *OTO – A DIY Platform for Mobile Social Robots in Education.* J. Adv. Intell. Syst. 630 (2018), 257-262.
[3] Filippov, S., Ten, N., Fradkov, A., Shirokolobov, I. *Robotics Education in Saint Petersburg Secondary School*. J. Adv. Intell. Syst. 630 (2018), 38-49.
[4] Islamgozhayev, T.U., Mazhitov, Sh.S., Zholmyrzayev, A.K., Toishybek, E.T. *IICT-bot: Educational robotic platform using omni-directional wheels with open source code and architecture.* Proceedings of IEEE International Siberian Conference on Control and Communications (SIBCON) (2015).
[5] Biyashev, R.G., Nyssanbayeva, S.E., Begimbayeva, Y., Magzom, M.M. Modification of the cryptographic algorithms, developed on the basis of nonpositional polynomial notations. *New Developments in Circuits, Systems, Signal Processing, Communications and Computers* (2015), 170-176.
[6] Gaba, D. 1999. *Human work environment and simulators*, In Anaesthesia 5, R.D. Miller, Ed. Churchill Livingstone, London, United Kingdom, 18–26.
[7] Staranowicz, A., Mariottini G.L. 2011. A survey and comparison of commercial and open-source robotic simulator software. *In Proceedings of the 4th International Conference on PErvasive Technologies Related to Assistive Environments* (PETRA '11). ACM, New York, NY, USA, Article 56, 8 pages.
[8] Kumar, A., Mittal, A., Arya, R., Shah, A., Garg, S., Kumar, R. 2017. Hardware in the loop based simulation of a robotic system with real time control and animation of working model. International Conference on Inventive Systems and Control (ICISC), Coimbatore (2017), 1-5.
[9] Raspberry Pi. (2018). Raspberry Pi 3 Model B - Raspberry Pi. [online] Available at: https://www.raspberrypi.org/products/raspberry-pi-3-model-b/ [Accessed 11 Mar. 2018].
[10]Store.arduino.cc. (2018). Arduino Mega 2560 Rev3. [online] Available at: https://store.arduino.cc/usa/arduino-mega-2560-rev3 [Accessed 11 Mar. 2018].
[11] Asama, H., Sato, M., Bogoni, L., Kaetsu, H., Mitsumoto, A., Endo, I. Development of an omni-directional mobile robot with 3 DOF decoupling drive mechanism. *Proceedings of 1995 IEEE International Conference on Robotics and Automation* (1995).
[12] Diegel, O., Badve, A., Bright, G., Potgieter, J., Tlale, S. Improved mecanum wheel design for omni-directional robots. *Proceedings of Australian Conference on Robotics and Automation* (2002).
[13] Tătar, M.O., Popovici, C., Mândru, D., Ardelean, I., Pleşa, A. Design and development of an autonomous omni-directional mobile robot with Mecanum wheels. *Proceeding of 2014 IEEE International Conference on Automation, Quality and Testing, Robotics* (2014).