Design and development of a delta robot system to classify objects using image processing

Vo Duy Cong, Le Hoai Phuong
Industrial Maintenance Training Center, Ho Chi Minh City University of Technology (HCMUT), VNU-HCM, Ho Chi Minh City, Vietnam

Article Info

Article history:

Received Jul 22, 2022
Revised Sep 4, 2022
Accepted Sep 11, 2022

Keywords:

Arduino
Delta robot
Image processing
Raspberry Pi
Sorting system

ABSTRACT

In this paper, a delta robot is designed to grasp objects in an automatic sorting system. The system consists of a delta robot arm for grasping objects, a belt conveyor for transmitting objects, a camera mounted above the conveyor to capture images of objects, and a computer for processing images to classify objects. The delta robot is driven by three direct current (DC) servo motors. The controller is implemented by an Arduino board and Raspberry Pi 4 computer. The Arduino is programmed to provide rotation to each corresponding motor. The Raspberry Pi 4 computer is used to process images of objects to classify objects according to their color. An image processing algorithm is developed to classify objects by color. The blue, green, red (BGR) image of objects is converted to HSV color space and then different thresholds are applied to recognize the object's color. The robot grasps objects and put them in the correct position according to information received from Raspberry. Experimental results show that the accuracy when classifying red and yellow objects is 100%, and for green objects is 97.5%. The system takes an average of 1.8 s to sort an object.

Corresponding Author:

Vo Duy Cong
Industrial Maintenance Training Center, Ho Chi Minh City University of Technology (HCMUT)
268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Vietnam
Email: congvd@hcmut.edu.vn

1. INTRODUCTION

In recent decades, automation system has become more and more popular in both large and small factories [1]. Automatic machines are gradually replacing manual steps in the production process to increase productivity and product quality. A robot is one of the automatic devices to replace humans in the production process [2], [3].

Robots are increasingly being used in automation. With the ability to operate accurately and without fatigue, robots gradually replace humans in repetitive activities or in dangerous jobs [4]. Using robots in automation systems has helped increase productivity, reduce costs, reduce material waste, and reduce defective products. Some of the industrial tasks implemented by robot arms include pick and place, assembly lines, palletizing, welding, and cutting [5]–[11]. There are many types of robots developed for different requirements. Robots can be composed of prismatic joints or rotation joints. Robots with rotation joints are more popular due to their ability to create more flexible movements. The links of the robot can be connected in series or parallel structures. The parallel structure has the advantage of being more rigid with the small weight of the links, achieving great movement speed without vibration [12], [13]. Delta robot is a parallel robot that is most used in industry. Another type of robot with the advantages of lightweight, low inertia and energy consumption, and high-speed operation is a flexible manipulator [14]–[17]. This type of robot is widely applied in industrial applications and space robots. However, due to the flexible dynamics, distributed parameters and nonlinear nature of the system, dynamic modeling, and controller design of a flexible manipulator are challenging tasks [15]. So, developing a delta robot is more straightforward than a flexible manipulator.

In recent years, computer vision technology has been developed with many algorithms for image processing, especially detection and classification algorithm. These algorithms can be integrated into robot systems for many applications in the industry [18]. With the help of computer vision, the robot becomes more flexible, responds to changes in the environment, and perceives the surroundings more clearly, thereby providing more accurate operations. Integrating computer vision systems brings many benefits such as increased productivity, increased accuracy and quality, reduced material waste, and reduced product costs.

Information from a computer vision system is obtained from one or more cameras. The data is then processed by a central processor (usually a computer with powerful computing capabilities). The necessary information will be extracted and sent to the robot's controller. The extracted information can be color, shape, texture, histogram, 2D coordinates, or 3D coordinates [19]. Some common applications of robots with vision systems are dynamic grasping, automatic sorting, visual servoing, vision guide robot, and vision-based inspection [20]–[27].

In this paper, we develop an automatic sorting system using machine vision. The components of the system are a delta robot arm for grabbing objects, a belt conveyor for transmitting objects, a camera mounted above the conveyor to capture images of objects, and a computer for processing images to classify objects. An image processing algorithm is developed to classify objects by color. The blue, green, red (BGR) image of objects is converted to HSV color space and then different thresholds are applied to recognize the object’s color. The computer sends a command to the robot when it detects an object. The robot will move the object to the position according to the command received.

2. RESEARCH METHOD AND MATERIALS

This section describes the design and development of the sorting system using a delta robot to sort products according to their color. The system is designed to work as follows: the products are transmitted on a conveyor belt to the position of the camera placed above it; the camera captures the image of the products; a computer processes the image to sort it by color and determine its position on the conveyor; the computer then sends the processed information including the coordinates and color of the object to a robot; the robot will grasp the object on the conveyor and bring it to the correct location according to its color.

Figure 1 show the mechanical system designed on the SolidWork software. The robot system consists of three main parts: a delta robot arm with a soft gripper to pick and place products, a conveyor belt to transmit the product and an aluminum frame to mount all parts. The mechanical system plays an important role in the operation of the system, helping the system to operate correctly and stably. Therefore, designing the mechanical system is an important and essential process. The different parts of a mechanical system are designed separately and then assembled.

Figure 2 shows the complete design of the delta robot arm. This prototype of the delta robot arm has 4 degrees of freedom, the fourth being mounted on the moving plate and providing the rotation motion around the vertical axis for the end-effector. The moving platform is kept parallel to the fixed base during motion. It is connected to the base by three identical kinematic chains having an R-(RR)-(RR) architecture [28]. The parallel chains are actuated by the revolute joints, which are close to the base, using direct current (DC) servo motors fixed to the base. The base is an aluminum plate with a thickness of 10mm that carries the entire weight of the robot. Three DC servo motors are connected to three main arms. The main arms connect to the mobile base via linked arms and universal joints. The main arms are cut from stainless steel pipes and the linked arm are carbon fiber rods. With the light weight of the arms, the delta robot can provide high acceleration and speed capability. The DC servo motors used in this prototype are Planetary GP36 motors shown in Figure 3. This motor consists of a planetary gearbox with a ratio of 1:14 and an encoder with a resolution of 500 pulses per revolution. It operates with a maximum voltage of 12 V and a consumption current of 3 A. The mobile platform mounts a Nema 17 stepper motor as shown in Figure 4. The stepper motor is connected to a soft gripper.

The design of the conveyor belt is shown in Figure 5. The main frame of the conveyor is an aluminum frame with a size of 30×90 mm. This frame is used to mount four stands, an active roller, a passive roller, and a DC motor. The DC motor transmits the motion to the active roller through a timing belt transmission with a ratio of 1:2. In addition, the conveyor is also designed with a belt tensioner as shown in Figure 6.

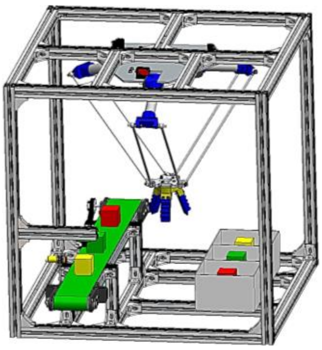

Figure 1. Robot system designed on the SolidWork software

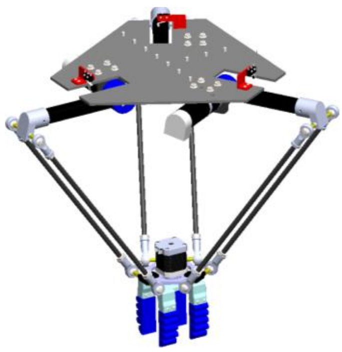

Figure 2. Delta robot arm

Figure 3. GP36 DC servo motor

Figure 4. Stepper motor

Figure 5. Structure of the belt conveyor

Figure 6. The belt tensioner

Figure 7 shows the wiring circuit of the controller using Arduino Mega and Figures 8(a) to 8(e) shows some main electronic devices used to build the circuit. The Arduino board creates signals and sends them to drivers to control the motor. The signals to control the DC servo motors and the stepper motor are pulse/direction signals. The driver for the DC servo motor is CC-SMART MSD_E10 driver Figure 8(b). It can be operated with a voltage from 10 to 40 V, a maximum amperage of 10 A, and maximum consumption power of 200 W. It is integrated with a PID controller and can use to control the position, velocity, or acceleration of the motors. The stepper motor is controlled via a Microstep driver with the smallest micro-step of 1/32 Figure 8(c). To turn on and turn off the DC motor for running the conveyor, a relay is used to convert the 5 V signal from Arduino to 12 V Figure 8(d). A UART-RS485 module is used to convert the UART of Arduino to the industrial communication standard rs485 for communication with the Raspberry Pi computer Figure 8(e).

In order to develop a computer vision system for classifying objects on the conveyor, a Raspberry Pi Camera Module with a Sony IMX219 8-megapixel sensor is utilized for capturing images of objects. This camera has a focal length of 3.04 mm, an angle of view (diagonal) of 62.2 degrees, and a resolution of up to 3280x2464 pixels. In the project, we only use images with a resolution of 640×480 pixels to achieve faster processing speed. The images are processed by Raspberry Pi 4 computer to classify objects according to their color. Figure 9 shows the flow chart of the image processing algorithm on the Raspberry Pi. The image taken by the camera is converted into hue, saturation, value (HSV) color space. The hue in HSV represents the color, saturation in HSV represents the greyness, and value in HSV represents the brightness. The HSV color space provides better performance when compared to RGB color space and hence it is used widely in the area of computer vision [29], [30]. To convert the BGR image to the HSV image, we can first define m and M as (1).

$$M = \max\{R, G, B\}\ ;\ m = \min\{R, G, B\} \quad (1)$$

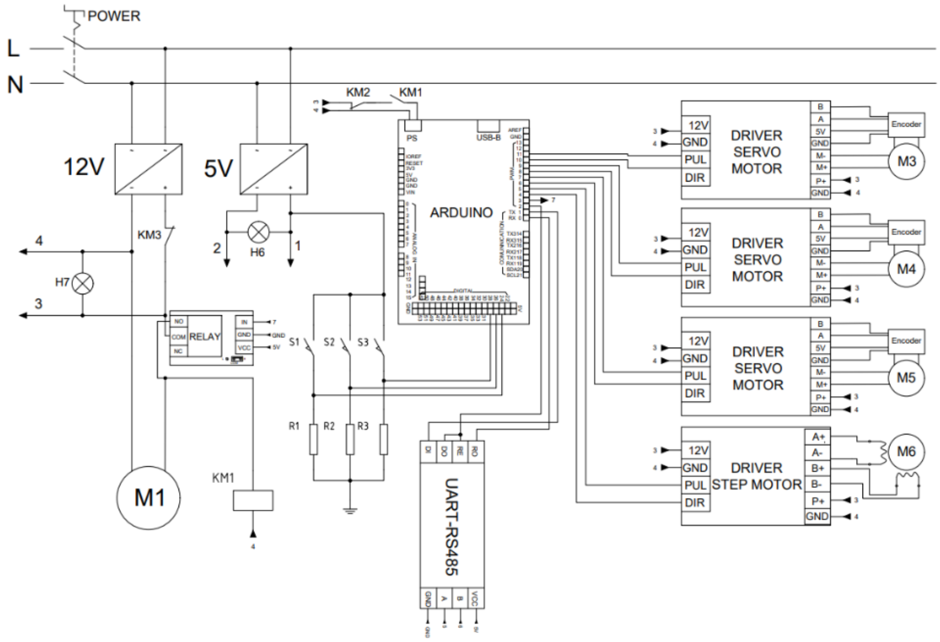

Figure 7. Wiring circuit of the controller

(a) (b) (c) (d) (e)

Figure 8. Some devices in the controller: (a) Arduino Mega, (b) CC-Smart MSD_E10 driver, (c) relay 5 V, (d) stepper motor driver, and (e) UART-RS485 module

The V and S are defined by the (2),

$$V = M/255$$
$$S = \begin{cases} 1 - m/M & if \quad M > 0 \\ 0 & if \ M = 0 \end{cases} \tag{2}$$

and the H value is defined by (3).

$$H = cos^{-1}\frac{R-(B+G)/2}{\sqrt{R^2+G^2+B^2-RG-GB-BR}} \qquad if\ G \geq B, or$$
$$H = 360^0 - cos^{-1}\frac{R-(B+G)/2}{\sqrt{R^2+G^2+B^2-RG-GB-BR}} \qquad if\ G < B \tag{3}$$

To detect objects with a specific color, we apply an upper bound and lower bound range for a range of each color in HSV. The results image is determined as (4).

$$\begin{aligned} dst(I) = {} & lowerb(I)_H \leq src(I) \leq upper(I)_H \wedge \\ & lowerb(I)_S \leq src(I) \leq upper(I)_S \wedge \\ & lowerb(I)_V \leq src(I) \leq upper(I)_V \end{aligned} \tag{4}$$

where *src(I)* is the source image, *dst(I)* is the destination image, $[lowerb(I)_H, upper(I)_H]$ is the range of hue, $[lowerb(I)_S, upper(I)_S]$ is the range of saturation, $[lowerb(I)_V, upper(I)_V]$ is the range of value. In this research, objects have red, green, or yellow colors. Firstly, the lower threshold and upper threshold for a range of red colors are applied. The resulting image is a binary image, if the objects are red, they will be white blobs in the binary image. If the area of the blob is larger than a threshold, it means that there is a red object. So, the Raspberry Pi will send a command for the red color to the delta robot. After that, the thresholds for the green and yellow objects are applied. If a green object or a yellow object is detected, the Raspberry Pi will send specific commands.

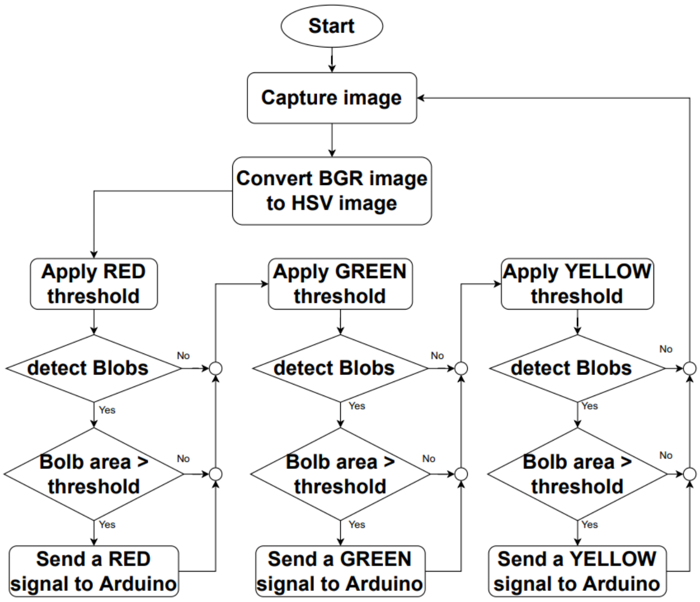

Figure 9. Flow chart of the image processing algorithm

If Arduino receives a color command from the Raspberry Pi, it will control the delta robot from the initial position to the conveyor to grasp the object. The robot will move the object to the position according to the command received. Then, the robot will return to the initial position and waits until a new order is received allowing the process to repeat in a continuous loop until the power is turned off.

3. RESULTS AND DISCUSSION

After the design is complete, the robot system is fabricated and assembled, the actual system is shown in Figure 10. Actual tests are performed to show that the model can work stably and classify objects correctly. In the actual system, Raspberry Pi is connected to the start/stop pushbuttons and the status indicator lights and housed in one electrical cabinet. The system will start working when the START button is pressed. The delta robot goes to the home position. The delta robot moves to the home position when it receives a START command from Raspberry Pi.

Figure 10. Real system

Experimentally, the thresholds for recognizing the colors of objects are as follows.

Red objects:
$[lowerb(I)_H, lowerb(I)_S, lowerb(I)_V] = [173, 0, 0]$
$[upperb(I)_H, upperb(I)_S, upperb(I)_V] = [255 ,255 ,255]$

Green objects:
$[lowerb(I)_H, lowerb(I)_S, lowerb(I)_V] = [58,155,114]$
$[upperb(I)_H, upperb(I)_S, upperb(I)_V] = [85,255,255]$

Yellow objects:
$[lowerb(I)_H, lowerb(I)_S, lowerb(I)_V] = [18, 0, 76]$
$[upperb(I)_H, upperb(I)_S, upperb(I)_V] = [36,255,255]$

The results of testing the thresholds are shown in Figure 11. It can be seen that the objects are extracted clearly in the binary image when the thresholds are applied. Figure 12 shows the results of recognizing many objects. The objects have been detected without confusion or omission. After detecting the object, the Raspberry Pi sends a signal to the Arduino to control the robot to pick up and drop the object in the correct position. Table 1 shows the classification results when applying the proposed method to classifying objects. For yellow and red products, there is no misclassification. For green objects, because the color of the objects is close to the background color, the classification results have some errors. The accuracy of the green objects achieved is 97.5%. Experimental results show that the system takes about 1.8 s to sort an object (including computer processing time and robot movement time).

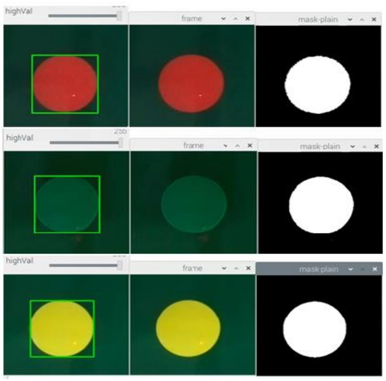

Figure 11. Testing thresholds

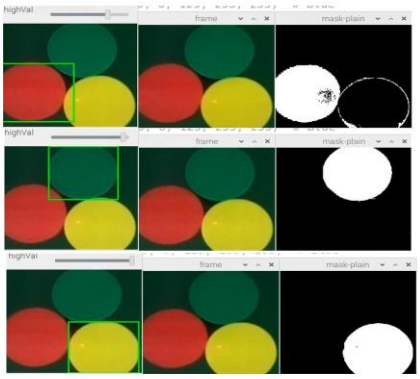

Figure 12. Classification results

Table 1. Classification results

Objects	No. of objects ‘image	Accuracy	Misclassification
Red	80	100%	0
Yellow	80	100%	0
Blue	80	97.5%	2

4. CONCLUSION

This paper has presented the design and development of a sorting system using a delta robot and computer vision. The delta robot is designed with lightweight materials for fast movement. Stepper motors and DC servo motors are used for precise and stable operation. The controller is designed based on the Arduino Mega board to output pulses to control the motors. The moving object on the conveyor is captured by the camera, the image is processed by the Raspberry Pi 4 computer.

An image processing algorithm is developed to classify objects by color. The BGR image is converted to HSV color space and then different thresholds are applied to recognize the object’s color. The experiment results have shown that the algorithm works well to classify objects without confusion or omission.

Experimental results have tested the effectiveness of the design. The system can be applied in industries, replacing humans in automatic product classification. The system can be extended to classify products according to different characteristics such as shape, size, material, and weight.

ACKNOWLEDGEMENTS

We acknowledge Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for supporting this study.

REFERENCES

[1] V. Cong, "Industrial robot arm controller based on programmable system-on-chip device," *FME Transactions*, vol. 49, no. 4, pp. 1025–1034, 2021, doi: 10.5937/fme2104025C.
[2] M. Zia Ur Rahman, M. Usman, A. Farea, N. Ahmad, I. Mahmood, and M. Imran, "Vision-based mobile manipulator for handling and transportation of supermarket products," *Mathematical Problems in Engineering*, vol. 2022, pp. 1–10, Jun. 2022, doi: 10.1155/2022/3883845.
[3] V. Cong, L. Hanh, L. Phuong, and D. Duy, "Design and development of robot arm system for classification and sorting using machine vision," *FME Transactions*, vol. 50, no. 2, pp. 181–181, 2022, doi: 10.5937/fme2201181C.
[4] H. M. Ali, Y. Hashim, and G. A. Al-Sakkal, "Design and implementation of Arduino based robotic arm," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 2, pp. 1411–1418, Apr. 2022, doi: 10.11591/ijece.v12i2.pp1411-1418.
[5] C.-S. Chen, S.-K. Chen, C.-C. Lai, and C.-T. Lin, "Sequential motion primitives recognition of robotic arm task via human demonstration using hierarchical BiLSTM classifier," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 502–509, Apr. 2021, doi: 10.1109/LRA.2020.3047772.
[6] P. Sutyasadi and M. B. Wicaksono, "Joint control of a robotic arm using particle swarm optimization based H2/H∞ robust control on Arduino," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 2, pp. 1021–1029, Apr. 2020, doi: 10.12928/telkomnika.v18i2.14749.
[7] A. H. Basori, "End-effector wheeled robotic arm gaming prototype for upper limb coordination control in home-based therapy," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 4, pp. 2080–2086, Aug. 2020, doi: 10.12928/telkomnika.v18i4.3775.
[8] W.-C. Chang, M.-Y. Cheng, and H.-J. Tsai, "Image feature command generation of contour following tasks for SCARA robots employing image-based visual servoing—A PH-spline approach," *Robotics and Computer-Integrated Manufacturing*, vol. 44, pp. 57–66, Apr. 2017, doi: 10.1016/j.rcim.2016.08.002.
[9] L. D. Hanh and V. D. Cong, "Implement contour following task of objects with unknown geometric models by using combination of two visual servoing techniques," *International Journal of Computational Vision and Robotics*, vol. 12, no. 5, pp. 464–486, 2022, doi: 10.1504/IJCVR.2022.10048788.
[10] R. Smith, E. Cucco, and C. Fairbairn, "Robotic development for the nuclear environment: Challenges and strategy," *Robotics*, vol. 9, no. 4, Nov. 2020, doi: 10.3390/robotics9040094.
[11] C. Weckenborg, K. Kieckhäfer, C. Müller, M. Grunewald, and T. S. Spengler, "Balancing of assembly lines with collaborative robots," *Business Research*, vol. 13, no. 1, pp. 93–132, Apr. 2020, doi: 10.1007/s40685-019-0101-y.
[12] A. Raoofian, A. Taghvaeipour, and A. K. E., "On the stiffness analysis of robotic manipulators and calculation of stiffness indices," *Mechanism and Machine Theory*, vol. 130, pp. 382–402, Dec. 2018, doi: 10.1016/j.mechmachtheory.2018.08.025.
[13] M. Wu, J. Mei, Y. Zhao, and W. Niu, "Vibration reduction of delta robot based on trajectory planning," *Mechanism and Machine Theory*, vol. 153, Nov. 2020, doi: 10.1016/j.mechmachtheory.2020.104004.
[14] G. Pang, J. Deng, F. Wang, J. Zhang, Z. Pang, and G. Yang, "Development of flexible robot skin for safe and natural human–robot collaboration," *Micromachines*, vol. 9, no. 11, Nov. 2018, doi: 10.3390/mi9110576.
[15] A. N. Kasruddin Nasir, M. A. Ahmad, and M. O. Tokhi, "Hybrid spiral-bacterial foraging algorithm for a fuzzy control design of a flexible manipulator," *Journal of Low Frequency Noise, Vibration and Active Control*, vol. 41, no. 1, pp. 340–358, Mar. 2022, doi: 10.1177/14613484211035646.
[16] D. Feliu-Talegon, V. Feliu-Batlle, I. Tejado, B. M. Vinagre, and S. H. HosseinNia, "Stable force control and contact transition of a single link flexible robot using a fractional-order controller," *ISA Transactions*, vol. 89, pp. 139–157, Jun. 2019, doi: 10.1016/j.isatra.2018.12.031.
[17] M. G. Alkalla and M. A. Fanni, "Integrated structure/control design of high-speed flexible robot arms using topology optimization," *Mechanics Based Design of Structures and Machines*, vol. 49, no. 3, pp. 381–402, Apr. 2021, doi: 10.1080/15397734.2019.1688170.
[18] W. T. Abbood, O. I. Abdullah, and E. A. Khalid, "A real-time automated sorting of robotic vision system based on the interactive design approach," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 14, no. 1, pp. 201–209, Mar. 2020, doi: 10.1007/s12008-019-00628-w.
[19] Z. Wang, H. Li, and X. Zhang, "Construction waste recycling robot for nails and screws: Computer vision technology and neural network approach," *Automation in Construction*, vol. 97, pp. 220–228, Jan. 2019, doi: 10.1016/j.autcon.2018.11.009.
[20] S. Chen, Y. Li, and N. M. Kwok, "Active vision in robotic systems: A survey of recent developments," *The International Journal of Robotics Research*, vol. 30, no. 11, pp. 1343–1377, Sep. 2011, doi: 10.1177/0278364911410755.
[21] B. Jia, S. Liu, and Y. Liu, "Visual trajectory tracking of industrial manipulator with iterative learning control," *Industrial Robot: An International Journal*, vol. 42, no. 1, pp. 54–63, Jan. 2015, doi: 10.1108/IR-09-2014-0392.
[22] F. Zha *et al.*, "Semantic 3D reconstruction for robotic manipulators with an eye-in-hand vision system," *Applied Sciences*, vol. 10, no. 3, Feb. 2020, doi: 10.3390/app10031183.
[23] V. D. Cong and L. D. Hanh, "A new decoupled control law for image-based visual servoing control of robot manipulators," *International Journal of Intelligent Robotics and Applications*, vol. 6, no. 3, pp. 576–585, Sep. 2022, doi: 10.1007/s41315-022-00223-5.
[24] D. Ristic-Durrant, S. M. Grigorescu, A. Graser, Z. Cojbasic, and V. Nikolic, "Robust stereo-vision based 3D object reconstruction for the assistive robot friend," *Advances in Electrical and Computer Engineering*, vol. 11, no. 4, pp. 15–22, 2011, doi: 10.4316/AECE.2011.04003.
[25] A. Zakhama, L. Charrabi, and K. Jelassi, "Intelligent selective compliance articulated robot arm robot with object recognition in a multi-agent manufacturing system," *International Journal of Advanced Robotic Systems*, vol. 16, no. 2, Mar. 2019, doi: 10.1177/1729881419841145.
[26] H. Wei, X.-X. Chen, and X.-Y. Miao, "Vision-guided fine-operation of robot and its application in eight-puzzle game," *International Journal of Intelligent Robotics and Applications*, vol. 5, no. 4, pp. 576–589, Dec. 2021, doi: 10.1007/s41315-021-00186-z.

[27] O. Abdullah, W. Abbood, and H. Hussein, "Development of automated liquid filling system based on the interactive design approach," *FME Transactions*, vol. 48, no. 4, pp. 938–945, 2020, doi: 10.5937/fme2004938A.
[28] J. Somlo, G. D. Varga, M. Zenkl, and B. Mikó, "The 'Phantom' delta robot a new device for parallel robot investigations," *Acta Polytechnica Hungarica*, vol. 15, no. 4, pp. 143–160, 2018, doi: 10.12700/APH.15.4.2018.4.8.
[29] V. Chernov, J. Alander, and V. Bochko, "Integer-based accurate conversion between RGB and HSV color spaces," *Computers & Electrical Engineering*, vol. 46, pp. 328–337, Aug. 2015, doi: 10.1016/j.compeleceng.2015.08.005.
[30] M. Intisar, M. Monirujjaman Khan, M. Rezaul Islam, and M. Masud, "Computer vision based robotic arm controlled using interactive GUI," *Intelligent Automation & Soft Computing*, vol. 27, no. 2, pp. 533–550, 2021, doi: 10.32604/iasc.2021.015482.

BIOGRAPHIES OF AUTHORS

Vo Duy Cong received his M.Sc. degree in mechatronics engineering from Ho Chi Minh City University of Technology in 2021. He joined the Industrial Maintenance Training Center, Ho Chi Minh City University of Technology, Vietnam National University-Ho Chi Minh City, in 2018. His research interests are mechanical design, LQR control, LQG control, model predictive control, industrial robot, mobile robot, visual servoing control, system modeling, computer vision, embedded system programming, automation system, and industrial maintenance. He can be contacted at congvd@hcmut.edu.vn.

Le Hoai Phuong received her Ph.D. degree in robotics from Ritsumeikan University, Japan, in 2013. From 2013 to 2015, she became a postdoctoral fellow postdoctoral researcher at Ritsumeikan Create Innovation in Industry Program (R-CIP) at the same University. During her postdoctoral period, she joined three months internship program as a postdoctoral fellow at Mitsubishi Electronic company in Japan. From 2015 to present, she is a lecturer at Industrial Maintenance Training Center, Ho Chi Minh City University of Technology, Vietnam. She is a founder of Viet-robot Center (now Rihime.Lab) at 10, 1 Street, Gia Hoa, Phuoc Long B Ward, District 9, Ho Chi Minh City, Vietnam. Her current research interests include soft robotics, automation engineering, micro-part feeding, vibration Feeding, vision System for industrial robots, aerial robots, STEM education for kids, and R&D technology toys as well as building educational models for kids. She can be contacted at lhphuong@hcmut.edu.vn.