Control the Robot Arm through Vision-Based Human Hand Tracking

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

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

In this paper, hand tracking based on computer vision is developed to control the movement of a SCARA robot arm. The robot arm will move according to the movement of the human hand. Instead of using buttons on the teach-pendant or a computer control program to move the robot arm, the robot can now be easily controlled and positioned quickly by the movement of the operator's hand. A SCARA robot arm with two rotation joints and one translation motion is constructed for the validation system. Two states of the hand are recognized for controlling the vacuum cup to grasp the products. Stepper motors drive the robot arm. Arduino Uno is used as the main controller for controlling the stepper motors. The hand-tracking is performed by using the MediaPipe Hands framework developed by Google. The coordinates of 21 hand landmarks are extracted for further processing. A program is written on a personal computer to process the image to get the position and state of the hand. This position is transformed into the rotation angles of the robot's joints. Then, the angles and state are sent to the Arduino board. The Arduino board creates pulse signals to rotate the stepper motors. The experimental results show that the robot's trajectory is close to the hand trajectory at a low speed.

Keywords: *hand tracking, SCARA robot, computer vision, Arduino, MediaPipe.*

1. INTRODUCTION

In recent decades, using human workers in manufac–turing and industries has been replaced by automated machines and robots. Robots appear increasingly in fac–tories, enterprises, and workshops to take on many different jobs [1-5]. To work in an automation system, a robot is often programmed to perform a repetitive task. The programming for robot operation can be done on a teach-pendant or use the software on a computer. Nor–mally, robots will be simulated on computer software to be able to check that the robot's operation is correct, in accordance with requirements, and avoid collisions [6-7]. This requires the simulation environment to be as close to the real environment as possible. Therefore, it takes time to set up the software to match the actual environment. To shorten this time, this paper proposes a method of controlling the robot through the movement of the human hand. The robot is operated daily in real conditions, so there is no setup time, and the operation is directly monitored, so errors are limited. The robot is operated directly in real conditions, thus reducing setup time and directly monitoring the operation, so mistakes are reduced.

There are many techniques for recognizing the hand motion to control the robot, such as Vision-based Gesture Recognition [8], Motion Capture Sensor Re–cognition [9], laser-based tracking system [11], Acce–lerometer-based Gesture Recognition [10], human arm movement detection using potentiometers [12], Body Sensor Network [13] … In [14], Zhou et al. used two wearable inertial sensors to measure human upper limb movement. A Kalman filter is applied to reduce the measurement drift in segment orientation. The accele–rometers and a gyroscope are also used to measure the rotation angle of the user's lower arm [15]. This angle is transmitted wirelessly to a controller for controlling an anthropomorphic robotic arm. Ha [16] developed a ro–botic hand to control movement with a glove. The flex sensors are used to measure the curvature of fingers. The data is transmitted to the Arduino board to control a glove unit by the nRF24l01 module. The information is processed to control five servo motors to move five fingers of the glove.

Computer vision is a powerful method for hand recognition and tracking. Powerful image processing al–gorithms can detect hand motion. The image of the human hand is processed to segment the hand from the stationary background and lighting conditions and to select features to represent gestures that enhance gesture classification accuracy [17-20]. In [18], Hand Recog–nition Software based on computer vision is developed to operate a robot arm. The robot moves according to the finger count to pick up and drop metallic objects. A robotic arm controlled by hand gesture was presented in [19]. The robot arm consists of 4-DOF, which are dri–ven by steeper motors. An algorithm based on the contour and convex hull technique is developed for hand gesture recognition. Haji Mohd et al. [20] used deep learning techniques for fast and robust hand detec–tion and tracking. An algorithm combines the Kerne–lized Correlation Filter (KCF) tracker with the Single-Shot Detection (SSD) method to overcome some challenges, such as cluttered backgrounds and occlu–

Received: August 2023, Accepted: November 2023

Correspondence to: Dr. Le Hoai Phuong
Industrial Maintenance Training Center,
Ho Chi Minh City University of Technology, Viet Nam
E-mail: lhphuong@hcmut.edu.vn

doi: 10.5937/fme2401037P

sions. In [21], a hybrid detection/reconstruction convo–lutional neural networks (CNN) framework is const–ructed to detect regions of hands. Then, the feature maps of region proposals are used to classify the hand and background, refine the position of hands, and re–construct the appearances of hands simultaneously. Wu et al. [22] trained a deep learning model based on a data-driven approach for a system to estimate a hand pose in 3D space. The training procedure used a ske–leton-difference loss function to consider the physical constraints of a hand and an object-manipulating loss function to consider the knowledge of the hand-object interaction.

Although hand detection and tracking algorithms have been developed a lot, the application of hand movements for direct control of the robot's position has not been developed. Some studies only use hand ges–tures to control the movements of robot arms or mobile robots. Conducted studies often do not pay attention to the accuracy of robot positioning. Robots are usually only required to move in directions depending on hand gestures. This method cannot control the motion trajectory of the end-effector robot. In real applications, the end-effector robot's trajectory must be determined accurately. However, creating a complex trajectory takes a lot of time. In this paper, a method for control–ling the movement of a robot arm based on hand move–ments is proposed. The robot's end-effector is controlled to move according to the coordinates of the human hand detected by the vision system. This makes it easier to directly control the robot's movements in the workspace. The proposed method can be used for practical appli–cations such as welding and laser cutting.

In this paper, the position of the human hand in the image is extracted to control the movement of a SCARA robot arm. A SCARA robot arm with two rotation joints and one translation motion is constructed for the validation system. Two states of the hand are recognized for controlling the vacuum cup to grasp the products. Stepper motors drive the robot arm. Arduino Uno is used as the main controller to control the stepper motors. A program is written on a personal computer to process the image to get the hand's position. This position is transformed into the rotation angles of the robot's joints. Then, the angles are sent to the Arduino board. The Arduino board creates pulse signals to rotate the stepper motors.

2. MATERIALS AND METHODS

2.1 Hand tracking

Visual hand tracking is a process that uses computer vision to detect the human hand and keep following the movement of the hand. Hand tracking is applied in many applications such as gesture recognition systems, virtual reality, Virtual Touchscreen, and human-com–puter interaction… The hand-tracking feature allows you to use natural hand movements to control, move, hold, and touch subjects without using bulky control–lers. In this paper, we use the position of the human hand to control the position of a robot arm. That means the robot arm will move according to the movement of the human hand. Instead of using buttons on the teach-pendant or a computer control program to move the robot arm, the robot can now be easily controlled and positioned quickly by the movement of the operator's hand.

Numerous techniques have been developed for hand tracking over several years. Simple techniques are often based on features such as skin color, contour, boundary, and shape. Skin color is a useful and robust method. However, it is very difficult to detect skin color under a variety of lighting and environments. Many machine-learning algorithms also have been proposed for hand tracking [23]. Trained models must ensure that they detect hands in the presence of noise and other skin-color regions and can run in real-time (fast enough).

Google developed the MediaPipe framework, which contains many machine-learning solutions. One of the solutions is the hand and finger tracking solution called MediaPipe Hands [24]. MediaPipe Hands is based on the BlazePalm CNN architecture that consists of two models working together: palm detection and landmark detection.

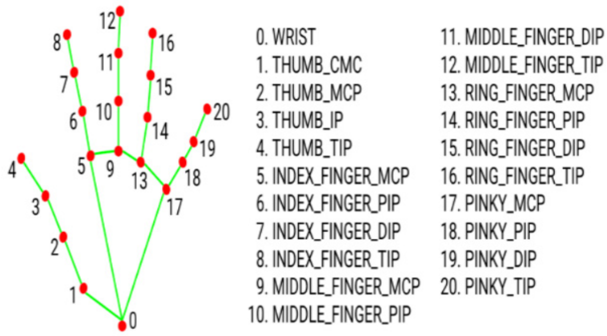


Figure 1. 21 hand landmarks in the detection result

Hand Palm Detector: locating hand region in the full input image. The result is oriented hand-bounding boxes that contain the hand palm.

Hand Landmarks model: estimating precise 2.5D landmarks by operating on the hand-bounding box. The hand landmark model can predict 21 precise coordinates of the location of each hand landmark. Figure 1 describes 21 hand landmarks in the detection result.

Figure 2 shows the result of hand detection. The hand landmarks are connected to each other by straight lines to draw the hand shape. The coordinates of the centroid of the triangle formed by the hand landmarks 0, 5, and 17 are calculated. A pink circle illustrates this point. The distance between the WRIST hand landmark and the MIDDLE_FINGER_TIP hand landmark is used to distinguish two states of the hand, opened and closed. When this distance is larger than a threshold, the hand state is opened state and vice versa.

2.2 Robot arm design and control

Figure 3 depicts the robot arm designed in the Inventor 2021 software. The robot has two degrees of freedom to move the end-effector in the X-Y plane. The stepper motors are mounted at each joint to provide rotation motion. A cylinder and a vacuum cup are attached to grasp objects at the end of the second link. The lengths of each link are 160mm and 140mm, respectively; the robot has a maximum reach of 300mm. In the first joint, the motor is directly connected to the robot arm (using a flanged shaft connection). However, in the second joint, the motor is not directly connected to the second link, and the movement from the motor is brought to the link via a toothed belt drive. This helps the second motor to be placed closer to the first joint, reducing the moment of inertia so that the first motor can accelerate faster.

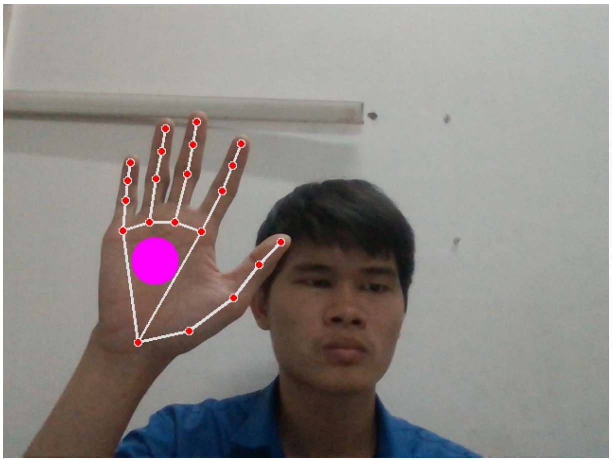


Figure 2. The result of hand detection

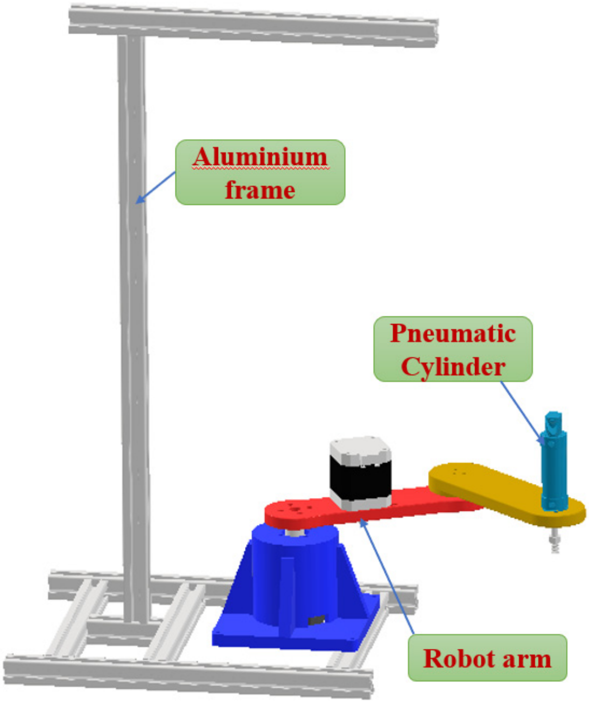


Figure 3. The robotic arm was designed on the Inventor software.

The robot links are fabricated by using a 3D printer machine. 3D printing is a simple technology that is being widely used for prototyping products in many applications. In recent years, 3D printing applications have become more accessible and cost-effective due to improvements and changes in printing technology.

The timing belt transmission is the GT2-20-5-6W model with the dimensions shown in Table 1. The pulley has a bore size of 5mm to assemble with the stepper motor shaft. The pulley consists of two 3mm set screws to fix the shaft.

The stepper motors are 2-phase hybrid stepper motors with a frame size of 42mm and a motor length of 55mm; the motor can provide a holding torque of 0.55 Nm. The shaft diameter is 5mm. The detailed speci–fication of the motor is depicted in Table 2.

Table 1. The timing belt transmission specification

Belt Width	6mm
Length of belt	200mm
No. of pulley's teeth	20 teeth
Pitch	2mm
Pulley material	Aluminum
Bore Size	5mm
Set screws included	3mm

Table 2. The stepper motor specification

Frame size	42x42mm
Motor length	55mm
Holding torque	0.55Nm
Steps/Revolution	200
Rated Current	2.2 A
Motor Weight	0.6kg
Winding Ohms	6.9mH
Rotor Inertia	220 g.cm^2

In this project, the Arduino Uno board is used as the main controller to control the robot. Arduino Uno creates the pulse signals and sends them to the stepper motor drivers to rotate the motors. Figure 4 shows the wiring circuit of the electronic system.

The electrical system consists of two motor drivers, two 5V relay modules, and two 5/2 valves that are controlled by an Arduino board. The stepper motor drivers have 12 pins: six pins are signal pins to connect with the microcontroller, four pins are connected to the stepper motor, and two pins for the power supply. The signal pins consist of EN+, EN-, DIR+, DIR-, PUL+, PUL-. In which EN+ and EN- are not used. DIR+ and DIR- pins are used to change the rotation direction. PUL+ and PUL- are pulse signals. The motor turns a micro-step when a pulse is sent to the driver. The micro-step can be changed by setting the state of switches 1, 2, and 3 on the driver. The chosen micro-step is 1, 2, 4, 8, 16, and 32. In addition, the maximum current can also be changed by the 4, 5, and 6 switches. Two relay modules, 5V, are used so that the Arduino board can control the 5/2 valve. The signal of Arduino pins has a voltage level of 5V, so it can't trigger valve 5/2 with a rated voltage of 24V. Therefore, an intermediate relay module is needed to convert the voltage level. The module relay has two pins for supplying a power of 5V, a signal pin. When the voltage sent to the signal pin changes, the state of the relay will change. The NO contact of the relay module is connected to the negative pin of the power supply and the coil of the 5/2 valve. The other coil pin will connect to the positive pin of the power supply. Therefore, current will flow from the 24V power supply through the relay contact and supply to the coil of the 5/2 valve. A 24V-5A power is used to supply the stepper motor driver and valve 5/2. Each stepper motor needs a current of 2.2A, so the total current for four motors is about 4.4 A.

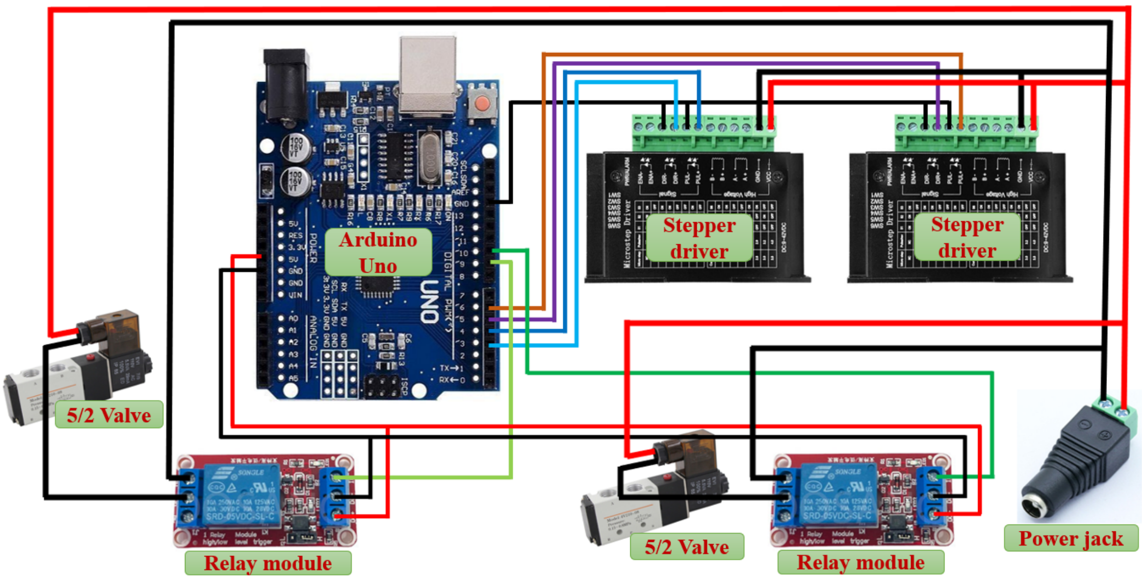


Figure 4. The wiring circuit.

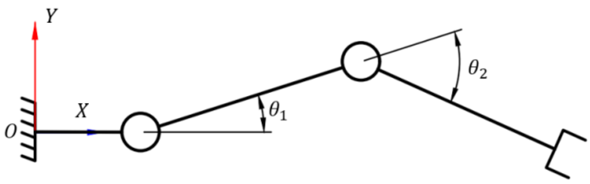


Figure 5. The schematic diagram of the robot arm.

Figure 5 is the schematic diagram of the robot arm. From Figure 5, the relationship between the robot joint angle and the position of the end-effector is depicted by equation (1):

$$\begin{aligned} x &= l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2) \\ y &= l_1 \sin\theta_1 + l_2 \sin(\theta_1 + \theta_2) \end{aligned} \tag{1}$$

where (x,y) is the coordinate of the end-effector, l_1, l_2 are the length of links, θ_1, θ_2 are the joint angles. Solving Equation (1), we can obtain the joint angles to move the robot to the desired position:

$$\theta_2 = \pm \arccos \frac{x^2 + y^2 - l_1^2 - l_2^2}{2 l_1 l_2} \tag{2}$$

$$\theta_1 = \pm \arccos \frac{x}{r} - \phi \tag{3}$$

with $$r_1 = \sqrt{(l_1 + l_2 \cos\theta_2)^2 + (l_2 \sin\theta_2)^2} \tag{4}$$

$$\varphi = a\tan(l_2 \sin\theta_2, l_1 + l_2 \cos\theta_2) \tag{5}$$

The stepper motor will move when the driver receives the control pulse from the Arduino. When a pulse is sent to the driver, the motor rotates a micro-step. Assuming that the motor needs n pulses to complete one revolution, the number of pulses required for the motor to rotate an angle θ is:

$$m = \frac{\theta n}{2\pi} \tag{6}$$

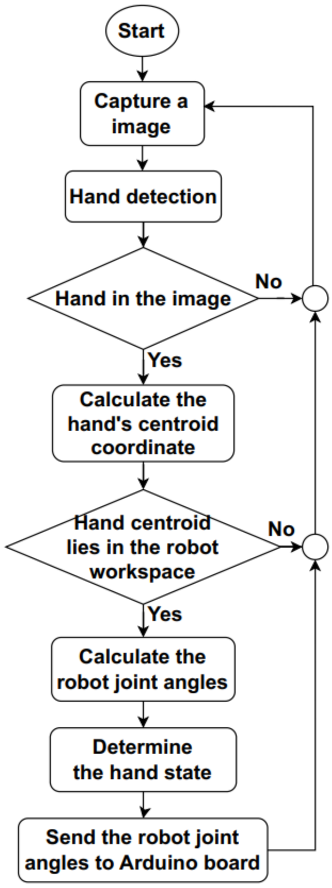


Figure 6. The flow chart of the hand tracking process.

Figure 6 is the flow chart to implement the hand tracking for robot control. Firstly, an image is captured by the camera. The MediaPipe Hands framework is applied to find and track the hand in the image. In the case of having a hand in the image, the hand landmarks are extracted, and the hand centroid is calculated. Then, if the centroid coordinates of the hand are in the robot workspace, the centroid is used to calculate the robot joint angles. The hand state is also determined. Finally, the robot joint angles and the hand state are sent to the Arduino board. The computer and the Arduino board communication is based on asynchronous serial com–munication. The baud rate is set to 9600 bps. The angles of rotation will be converted to a string before being sent. The holding and opening state of the hand is also encoded into two characters, '0' and '1'.

3. RESULTS AND DISCUSSION

Figure 7 shows the experimental model. The camera is placed above the robot arm. The resolution of the camera is 640x480 pixels. So, the pixel coordinates are divided by two to satisfy the robot workspace. The camera is connected to the computer by a USB port. A program is written in Python programming language to read and process the image. In Figure 8, two results of hand detection are presented. When the hand state opens, the two 5/2 valves are not activated. The cylinder is at the retreat position (the higher position). When the hand state is holding, the two 5/2 valves are activated, and the cylinder moves to the extended position to grip objects with the vacuum cup. In the resulting images, the centroid coordinates of the hand are extracted and represented by the pink circle. The two blue circles in the image represent the limits of the robot. When the hand is outside these two circles, the centroid coordi–nates of the hand will not be extracted, as shown in Figure 9. Test results have shown that the algorithm can detect the hand and determine the state of the hand when it is within the robot's working limits. Deter–mining the robot's operating range in the image helps the operator not move his hand to positions the robot cannot reach. The operator does not need to observe the robot's movements during operation.

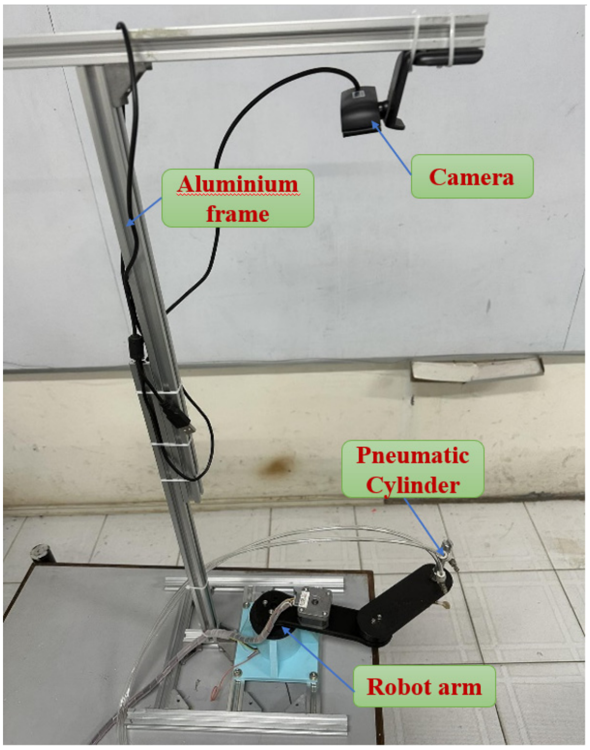
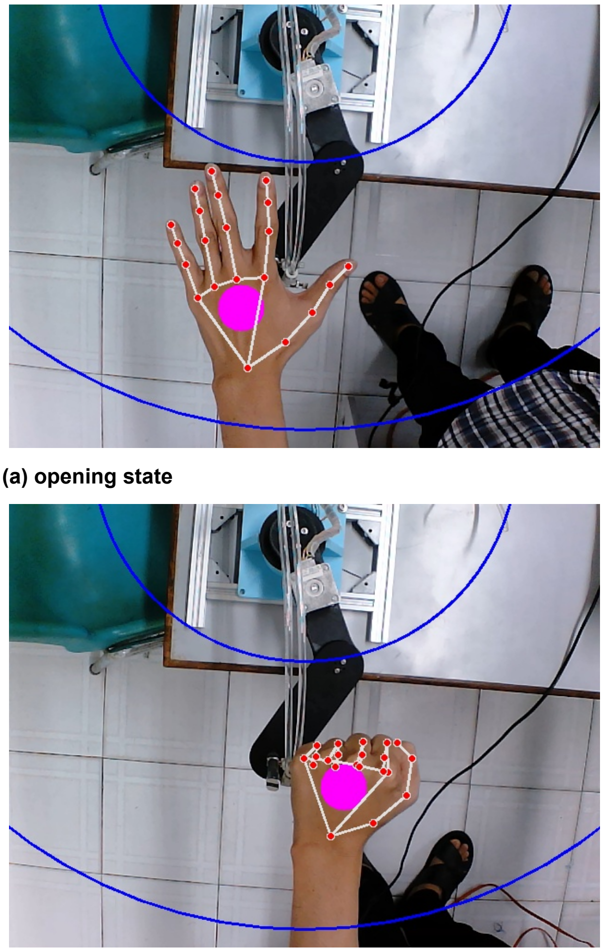


Figure 7. The experimental model.

The centroid coordinates of the hand are then used to calculate the rotation of the robot’s joints. The rotation angles are sent to the Arduino board and converted to the number of pulses of the stepper motor. To control stepper motors, the AccelStepper library is used.



(a) opening state

(b) closing state

Figure 8. The hand detection result with two states.

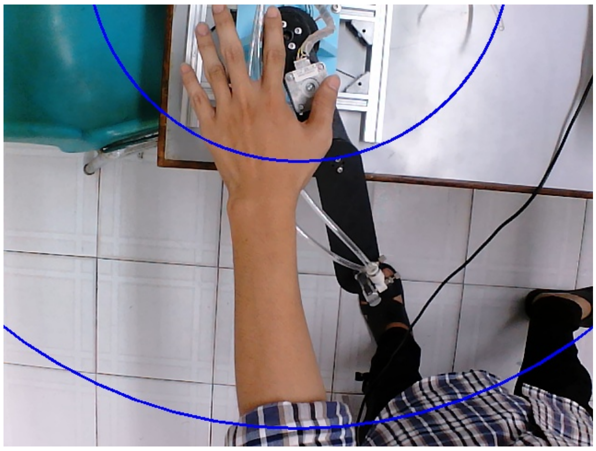
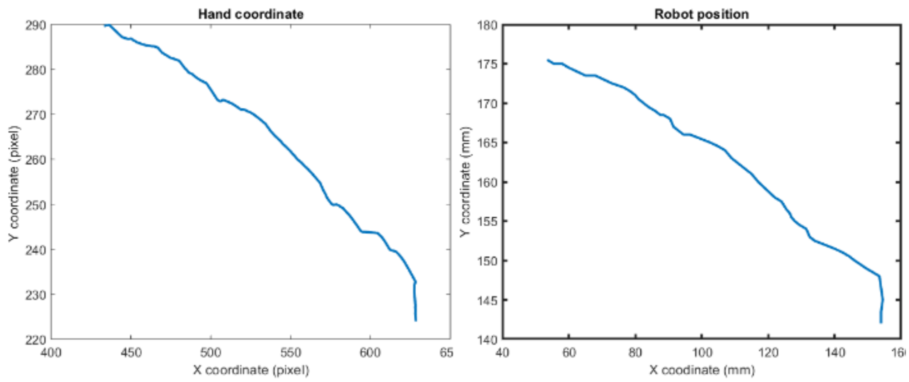
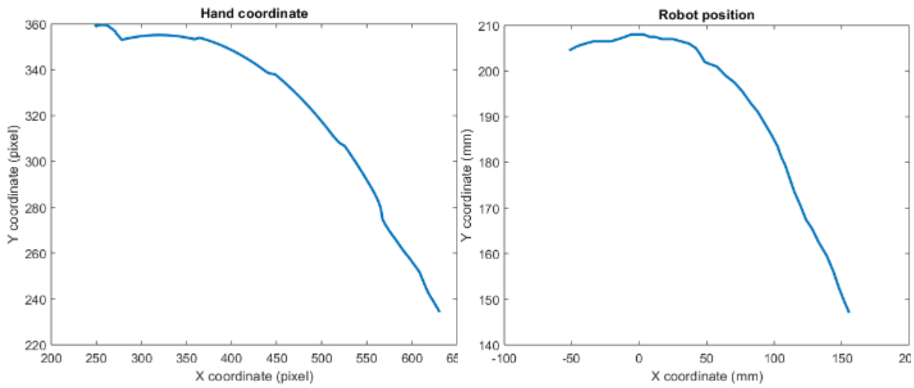


Figure 9. The hand is outside the robot workspace.

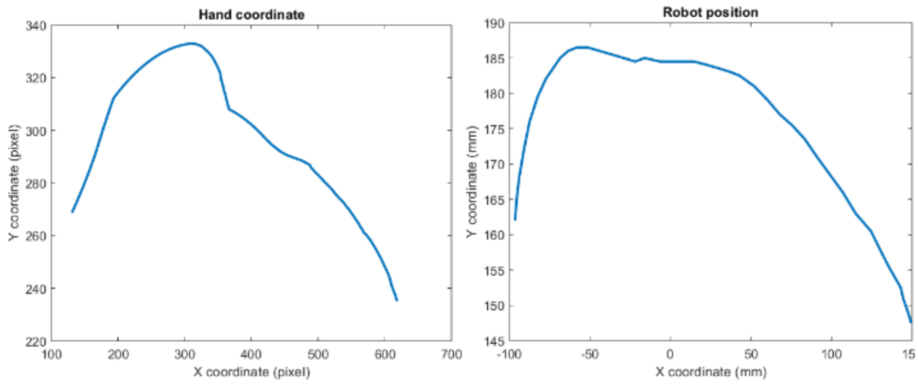
This library allows the control of stepper motors with acceleration and can control multiple motors si–multaneously. Figure 10 shows the trajectory of the hand in the image and the actual trajectory of the robot at various movement speeds. It can be seen that, at slow speed, the actual trajectory of the robot arm and the trajectory of the hand are similar. However, if the hand moves quickly, the robot arm will not be able to res–pond, so the two trajectories will be different. The limited speed of the stepper motor explains this.



(a) Low speed



(b) medium speed



(c) large speed

Figure 10. The trajectory of the hand in the image and the actual trajectory of the robot at various movement speeds.

4. CONCLUSION

This paper has developed a human hand-tracking system to control the position of a SCARA robot arm. The MediaPipe Hands framework is applied to find and track the hand in the image. The hand's centroid coordinates are calculated and transformed to the robot's joint angles. A SCARA robot arm is designed and fabri–cated for experiment validation. Stepper motors drive the robot arm, and the 3D printing technology is chosen for fabricating the robot parts. The Arduino Uno is used as the main controller to create pulse signals for stepper motor drivers. The stepper motors are controlled thanks to the AccelStepper library for smooth motion.

The robot's kinematics problem has been analyzed, thereby determining the robot's workspace. The work–space has been converted to the image space. Therefore, the range of motion of the human hand in the image is limited. During the control process, when the human hand goes out of bounds, the robot will not move, ensuring safety.

An experimental model has been built to test the feasibility of the proposed method. The robot has been fabricated, with the lengths of each link being 160mm and 140mm, respectively. So, the robot has a maximum reach of 300mm. The experimental results show that the states of the hand are precisely determined, and the robot moves exactly according to the trajectory of the hand at low speed. At high speed, the robot arm will not be able to respond to the speed of the hand, so the two trajectories will be different.

With the results achieved, in the future, we will apply the proposed method with an industrial robot arm in a real production system to shorten the initial setup time. Industrial cameras will be used to replace USB cameras to achieve higher capture rates, helping to increase response time for the tracking system.

ACKNOWLEDGMENT

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

REFERENCES

[1] Cong, V.D: Visual servoing control of 4-DOF palletizing robotic arm for vision-based sorting robot system, International Journal on Interactive Design and Manufacturing (IJIDeM), Vol. 17, pp. 717–728, 2023.

[2] Chen, X, Liu, P., Ying, R. and Wen, F.: Autonomous 6-DOF Manipulator Operation for Moving Targetby a Capture and Placement Control System, Sensors, Vol. 22, No. 13, 2022.

[3] Liu, Y., Wang, L., Makris, S., Krüger, J.: Smart robotics for manufacturing, Robotics and Computer-Integrated Manufacturing, Vol. 82, 2023.

[4] Luca, C., Emmanuel, F.: Safety 4.0 for Collaborative Robotics in the Factories of the Future, FME Transactions, Vol. 49, pp. 842-850, 2021.

[5] Polishchuk, M., Telenyk, S., Tkach, M.: Adaptive Assembly Module for Industrial Robot: Design and Simulation, FME Transactions, Vol. 50, pp. 149-157, 2022.

[6] D'Avella, S., Avizzano, C.A., Tripicchio, P.: ROS-Industrial based robotic cell for Industry 4.0: Eye-in-hand stereo camera and visual servoing for flexible, fast, and accurate picking and hooking in the production line, Robotics and Computer-Integrated Manufacturing, Vol. 80, 2023.

[7] D'Avella, S., Unetti, M. and Tripicchio, P.: RFID Gazebo-Based Simulator with RSSI and Phase Signals for UHF Tags Localization and Tracking, IEEE Access, vol. 10, pp. 22150-22160, 2022.

[8] Chen, R. and Tian, X.: Gesture Detection and Recognition Based on Object Detection in Complex Background, Applied Sciences, Vol. 13, No. 7, 2023.

[9] Perrin, S., Cassinelli, A. and Ishikawa, M.: Gesture recognition using laser-based tracking system, Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings., Seoul, Korea (South), 2004, pp. 541-546.

[10] Ducloux, J., Colla, P. Petrashin, P., Lancioni, W. and Toledo, L.: Accelerometer-based hand gesture recognition system for interaction in digital TV, 2014 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings, Montevideo, Uruguay, 2014, pp. 1537-1542.

[11] Bhatia, Y., Bari, A.H., Hsu, G-SJ., Gavrilova, M.:Motion Capture Sensor-Based Emotion Recognition Using a Bi-Modular Sequential Neural Network, Sensors, Vol. 22, No.1, 2022.

[12] Basil. Mathew, E., Khanduja, D., Sapra, B., Bhushan, B.: Robotic arm control through human arm movement detection using potentiometers, International Conference on Recent Developments in Control, Automation and Power Engineering (RDCAPE), 2015.

[13] Maheen, A. et al.: Human Hand Gesture Recognition System Using Body Sensor Network, International Conference on Robotics and Automation in Industry (ICRAI), 2021.

[14] Zhou H. and Hu, H.: Reducing Drifts in the Inertial Measurements of Wrist and Elbow Positions, in IEEE Transactions on Instrumentation and Measurement, Vol. 59, No. 3, pp. 575-585, March 2010.

[15] Sekhar, R., Musalay, R.K., Krishnamurthy, Y. and Shreenivas, B.: Inertial sensor based wireless control of a robotic arm, 2012 IEEE International Conference on Emerging Signal Processing Applications, Las Vegas, NV, USA, 2012, pp. 87-90.

[16] Ha, V.T., Thanh, N.T., Ha, V.T.: Design and fabrication of a moving robotic glove system, International Journal of Electrical and Computer Engineering, Vol. 13, No.3, pp. 2704-2710, 2023

[17] Gabriel, H. and Burkhard, W.: A Virtual Touchscreen with Depth Recognition, Proceedings of the Thirteenth Australasian User Interface Conference, Vol. 126, pp. 39-48, 2012.

[18] Çetinkaya, A.: Controlling A Robotic Arm Using Hand Recognition Software, International journal of engineering technologies, Vol. 5, No. 2, pp. 151-174, 2019.

[19] Zafar, S., Usman, M., Ahmad, F., Ahmad, N., Maqbool, H.F. and Imran, M.: Vision Based Gesture Controlled Robotic Arm, Proceedings of International Conference on Mechanical Engineering, pp. 533-539, 2021.

[20] Haji Mohd, M.N., Mohd Asaari, M.S., Lay Ping, O., Rosdi, B.A.: Vision-Based Hand Detection and Tracking Using Fusion of Kernelized Correlation Filter and Single-Shot Detection, Applied Sciences, Vol. 13, No. 13. 2023.

[21] Xu, C., Cai, W., Li, Y., Zhou, J., Wei, L.: Accurate hand detection from single-color images by reconstructing hand appearances, Sensors, Vol. 20, 2020.

[22] Wu, M., Ting, P., Tang, Y., Chou, E., Fu, L.: Hand pose estimation in object-interaction based on deep learning for virtual reality applications, J. Vis. Commun. Image Represent, Vol. 70, 2020.

[23] Ahmad, A., Migniot, C., Dipanda, A.: Hand pose estimation and tracking in real and virtual interaction. A review, Image Vis. Comput., Vol. 89, pp. 35–49, 2019.

[24] Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C.L., Grundmann, M.: Mediapipe hands: On-device real-time hand tracking, arXiv 2020, arXiv:2006.10214, 2020.

КОНТРОЛИШИТЕ РУКУ РОБОТА ПУТЕМ ПРАЋЕЊА ЉУДСКИХ РУКУ ЗАСНОВАНО НА ВИЗИЈИ

Л.Х. Фуонг, В.Д. Конг

У овом раду, праћење руку засновано на компју–терском виду је развијено за контролу кретања СЦАРА роботске руке. Рука робота ће се кретати у складу са кретањем људске руке. Уместо да користите дугмад на привеску за учење или компјутерски контролни програм за померање роботске руке, робот се сада може лако контро–лисати и брзо позиционирати покретом руке оператера. СЦАРА роботска рука са два ротирајућа зглоба и једним транслацијским покретом је конструисана за систем валидације. Два стања руке су препозната по контроли вакуумске чаше да би се ухватили производи. Корачни мотори покрећу руку робота. Ардуино Уно се користи као главни контролер за контролу корачних мотора. Праћење руку се врши коришћењем оквира МедиаПипе Хандс који је развио Гоогле. Координате 21 ручне оријентире се издвајају за даљу обраду. На персоналном рачунару је написан програм за обраду слике да би се добио положај и стање руке. Ова позиција се трансформише у углове ротације зглобова робота. Затим се углови и стање шаљу на Ардуино плочу. Ардуино плоча ствара импулсне сигнале за ротацију корачних мотора. Експери–ментални резултати показују да је путања робота блиска путањи руке при малој брзини.