DESIGN AND IMPLEMENTATION OF AN AI-INTEGRATED 6-DOF ROBOTIC ARM FOR AUTOMATED PACKAGE AND ORIENTATION RECOGNITION

Van Thuong Nguyen[1], Tien Ton That[1], Nhu Thanh Vo[1]*, Ba Hoa Thai[2]

[1]*The University of Danang - University of Science and Technology, Viet Nam*
[2]*Chungnam National University, Korea*

*Corresponding author: vnthanh@dut.udn.vn

(Received: April 30, 2025; Revised: June 08, 2025; Accepted: June 21, 2025)

DOI: 10.31130/ud-jst.2025.23(9D).555E

Abstract - This paper presents the design and fabrication of a 6-degree-of-freedom robotic arm, integrated with image processing for package recognition and artificial intelligence for determining the orientation of beverage cans. The robotic arm is constructed from extruded aluminum to ensure rigidity and flexibility, and it is precisely controlled by hybrid servo motors combined with a pulley-belt mechanism. The first camera detects the position and orientation of the can (standing or lying), while the second camera identifies the type of package moving along the conveyor. The system calculates the timing for the robot to pick up the correct number of cans corresponding to each package type (1, 2, or 3 compartments). Research and experimental results show that the package recognition system achieves an accuracy of 91.02%, while the AI model for determining can orientation reaches an accuracy of 99.375%, demonstrating high efficiency and reliability.

Key words - 6-degree-of-freedom robotic arm; hybrid servo motors; image processing; artificial intelligence

1. Introduction

The integration of robotics and artificial intelligence has driven transformative advances in many industries. This combination enables unprecedented levels of adaptability, precision, and human interaction in automated systems.

Research has demonstrated how image processing technology enhances a soft pneumatic actuator gripper, enabling more responsive manipulation of objects with diverse characteristics [1]. Additionally, studies show how Q-learning algorithms augmented with PID controllers improve the navigation accuracy of service robots in dynamic environments such as restaurants [2].

In the field of communication, research on robot speech using a cerebellum-like neural network to simulate human speech characteristics demonstrates a biomimetic approach to complex control problems [3-5]. Complementing these advances, modeling of delta robots using sophisticated simulation tools has provided valuable insights into the mechanical behavior of complex robotic structures [7], [8].

As these technologies continue to advance, they are collectively moving toward increasingly sophisticated autonomous systems capable of complex interactions with both the physical environment and humans, opening new frontiers for collaborative intelligence [9-13]. A 2020 report by the International Federation of Robotics indicates approximately 2.7 million industrial robots are currently in use worldwide, with McKinsey's 2022 Global Industrial Robotics Survey projecting that automation systems will account for 25% of capital expenditure across industrialized countries over the next five years [14-19]. This paper presents the design and fabrication of a 6-degree-of-freedom robotic arm constructed from extruded aluminum and controlled by hybrid servo motors with a pulley-belt mechanism, which integrates image processing for package recognition and artificial intelligence to determine beverage can orientation. The system uses dual cameras-one detecting can position and orientation, the other identifying package types (box) moving along the conveyor-to calculate precise timing for picking the correct number of cans corresponding to each package's compartment configuration.

2. Mechanical arm model

2.1. Model design

The 6-degree-of-freedom robotic arm was designed with lightweight, tensile-resistant, and durable materials, using aluminum profiles to ensure effective operation and mobility in harsh environments. Structurally, it features a flexible design adaptable to various applications. The base includes two sections: the lower part secures the robot to the ground, while the upper part connects to the motor and links to Joint 1. This joint is attached to the base plate with bolts and a rotational joint, allowing Axis 1 to rotate the robot left and right, expanding the working range up to 180 degrees. Joint 2 moves the lower arm forward and backward; Joint 3 extends the upper arm vertically; Joint 4 enables circular rotation for both horizontal and vertical movements. Joint 5 tilts the wrist up and down, and Joint 6, serving as the wrist itself, provides twisting motion for full rotational freedom.

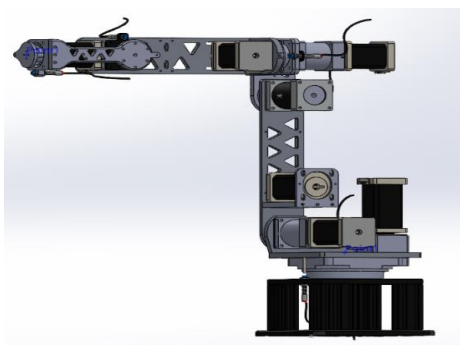

Figure 1. *Overall model of the robot arm in SolidWorks*

The detailed design of each link and joint is carried out methodically from top to bottom. The process begins by

identifying the most critical position-usually when the arm is fully extended horizontally-where bearing friction and gravitational moments are highest. A force distribution diagram is then drawn to calculate shaft moments. Motors are selected based on required torque, speed, and arm dimensions, with transmission ratios adjusted if necessary. The belt type is chosen accordingly. Finally, the components were modeled in SolidWorks, the belt length was determined, and appropriate materials were assigned to the objects in SolidWorks (for example, the robot arm frame was assigned aluminum profiles, and the motors were assigned materials to match their actual weights, etc.). In this configuration, when links 2, 4, 5, 6 are fully extended horizontally, the gravitational static moments on the axes reach their maximum values.

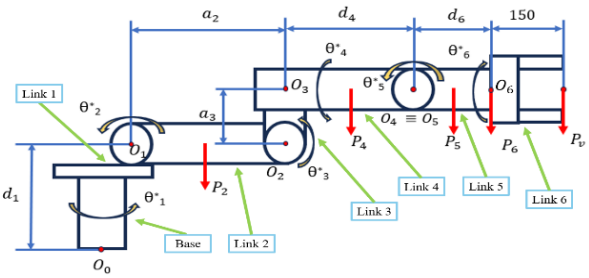

Figure 2. *Diagram of force acting on shaft*

Link 1 includes all mechanical parts from axis O_0 to O_1, and the same applies to links 2, 4, 5, and 6. Since the mass of link 1 does not affect the gravitational loads on the other links, it is not considered. Link 3 has very little mass and is mostly supported by link 2, so it is combined with link 2 and P_3 is set to 0 N.

Table 1. *Mass and gravitational force values at each link*

Component	Mass (g)	Gravitational Force (N)
Link 1 (m_1)	6937.37	-
Link 2 (m_2)	3924.47	38.46
Link 4 (m_4)	3203.62	31.395
Link 5 (m_5)	2756.80	27.017
Link 6 (m_6)	1840.69	18.039
Object (m_v)	1500	14.7

Table 2. *Values of parameters a_i and d_i*

Parameter	Value
a_1	52.25 mm
d_1	200 mm
a_2	268 mm
a_3	35 mm
d_4	326.4 mm
d_6	140 mm

Based on the gravitational force values in Table 1, the link lengths in Table 2, and the force distribution diagram acting on the shafts shown in Figure 2, the static moments caused by gravity on shafts O_1, O_2, O_3, O_4, O_5, and at the base are determined using the formula M = F*d [2] where M is the static moment, F is the applied force, and d is the moment arm (the distance from the center to the point of force application). For the base link, the distance from the center to the point of force application is 60 mm, which corresponds to the radius of the base bearing.

Table 3. *Static moment values acting on the shafts*

Parameter	Value
M_{t1}	68.043 Nm
M_{t2}	33.308 Nm
M_{t3}	9.574 Nm
M_{t4}	8.68 Nm
M_{t5}	1.703 Nm
M_{base}	11.856 Nm

Table 4. *Motor, gearbox, pulley-belt transmission system selection, and output torque values at corresponding shafts*

Shaft	Motor Model (Lichuan Co.)	Torque (Nm)	Gear Ratio	Belt Ratio	Output Torque (Nm)
O_1	LC57H2100	3	160:1	-	480
O_2	LC57H280	2.2	50:1	2:1	220
O_3	LC57H265	1.5	10:1	-	15
O_4	LC57H265	1.5	10:1	3:1	45
O_5	LC57H265	1.5	-	-	1.5
Base	LC57H2100	3	18:1	3:1	162

2.2. Forward kinematics calculation

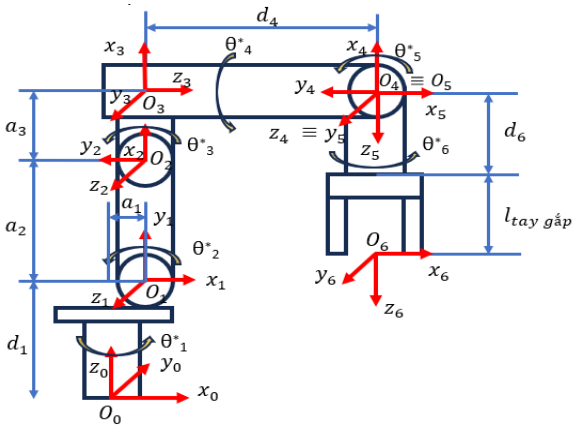

Figure 3. *Reference frame of the robot arm*

Based on Figure 3 and the above analysis, the Denavit-Hartenberg parameters of the robot can be determined as shown in Table 5.

Table 5. *Denavit-Hartenbergt (DH) table of robot arm kinematics problem*

Link	θ_i	α_i	a_i	d_i
1	θ^*_1	90°	a_1	d_1
2	θ^*_2	0°	a_2	0
3	θ^*_3	90°	a_3	0
4	θ^*_4	- 90°	0	d_4
5	θ^*_5	- 90°	0	0
6	θ^*_6	0	0	d_6

Convention: $cos\theta_1 = C_1, \quad sin\theta_1 = S_1,$

$$C_{23} = \cos(\theta_2 + \theta_3), S_{23} = \sin(\theta_2 + \theta_3).$$

We have the kinematic equations of the robot arm as follows [3]:

$$n_x = -C_1S_5C_6S_{23} - C_1S_4S_6C_{23} + C_1C_4C_5C_6C_{23} + S_1S_4C_5C_6 + S_1C_4S_6 \quad (1)$$

$$n_y = -S_1S_5C_6S_{23} - S_1S_4S_6C_{23} + S_1C_4C_5C_6C_{23} - C_1S_4C_5C_6 - C_1C_4S_6 \quad (2)$$

$$n_z = S_5C_6C_{23} - S_4S_6S_{23} + C_4C_5C_6S_{23} \quad (3)$$

$$o_x = -C_1S_4C_6C_{23} + C_1S_5S_6S_{23} - C_1C_4C_5S_6C_{23} - S_1S_4C_5S_6 + S_1C_4C_6 \quad (4)$$

$$o_y = -S_1S_4C_6C_{23} + S_1S_5S_6S_{23} - S_1C_4C_5S_6C_{23} + C_1S_4C_5S_6 - C_1C_4C_6 \quad (5)$$

$$o_z = -S_4C_6S_{23} - S_5S_6C_{23} - C_4C_5S_6S_{23} \quad (6)$$

$$a_x = C_1C_4S_5C_{23} + C_1C_5S_{23} + S_1S_4S_5 \quad (7)$$

$$a_y = S_1C_4S_5C_{23} + S_1C_5S_{23} - C_1S_4S_5 \quad (8)$$

$$a_z = C_4S_5C_{23} + C_5S_{23} \quad (9)$$

$$P_x = -d_4C_1S_{23} + d_6C_1C_5S_{23} + d_6C_1C_4S_5C_{23} + a_3C_1C_{23} + a_2C_1C_2 + S_1S_4S_5d_6 + a_1C_1 \quad (10)$$

$$P_y = d_4S_1S_{23} + d_6S_1C_5S_{23} + d_6S_1C_4S_5C_{23} + a_3S_1C_{23} + a_2S_1S_2 - C_1S_4S_5d_6 + a_1S_1 \quad (11)$$

$$P_z = -d_4C_{23} - d_6C_5C_{23} + d_6C_4S_5S_{23} + a_3S_{23} + a_2S_2 + d_1 \quad (12)$$

2.3. Inverse Kinematics

Inverse kinematics involves calculating joint angles based on a known end-effector position. To solve this, the end-effector's position and orientation must fall within the robot's reachable workspace.

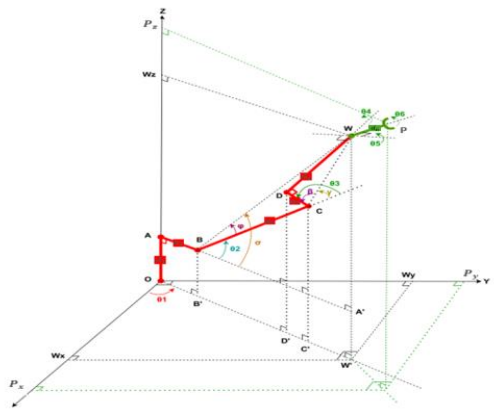

Figure 4. *Robot Geometry Diagram*

To solve the inverse kinematics problem, we divide it into two parts [3]:

- Part 1: Inverse Kinematics – Position: Given the working point P and the orientation of the end-effector (Matrix $^0R_6$ is a 3x3 rotation matrix that describes the orientation of the coordinate frame attached to the end-effector relative to the base frame), we can determine the wrist center (point W) using the equation:

$$[W_x, W_y, W_z]^T = [P_x, P_y, P_z]^T - d_6.[r_{31}, r_{32}, r_{33}]^T \quad (13)$$

In which:

The elements r_{31}, r_{32}, and r_{33} are the third row of matrix $^0R_6$ – typically corresponding to the z-vector of the end-effector.

P is the final end-effector position: P = $[P_x, P_y, P_z]^T$

d_6 is the length from the wrist to the end-effector.

W is wrist point position: W = $[W_x, W_y, W_z]^T$

This transforms the problem into an inverse kinematics problem for a 3-DOF robot arm, where the last three joint angles are neglected, allowing us to compute θ_1, θ_2, θ_3.

- Part 2: Inverse Kinematics – Direction: From the orientation matrix of the last link $^0R_6$ and the values θ_1, θ_2, θ_3 found in Part 1, we can find the matrix $^0R_3$. Next, we have a generalized matrix $^3R_6$. Using the two matrices $^0R_6$, $^0R_3$, and the generalized matrix $^3R_6$, we can calculate and determine the values of θ_4, θ_5, θ_6. The Spherical Wrist is a common mechanism in 6-DOF industrial robots, consisting of three revolute joints – Yaw (θ_4), Pitch (θ_5), and Roll (θ_6) – with their axes intersecting at a single point known as the spherical center.

We have

$$^0R_6 = {}^0R_3 \times {}^3R_6$$

$$^3R_6 = \begin{bmatrix} C_4C_5C_6 - S_4S_6 & -C_4C_5S_6 - S_4C_6 & C_4S_5 \\ S_4C_5C_6 + C_4S_6 & -S_4C_5S_6 + C_4C_6 & S_4S_5 \\ -S_5C_6 & S_5S_6 & C_5 \end{bmatrix}$$

$$^3R_6 = (^0R_3)^{-1} \times {}^0R_6 = \begin{bmatrix} C_1C_{23} & S_1C_{23} & S_{23} \\ S_1 & -C_1 & 0 \\ C_1S_{23} & S_1S_{23} & -C_{23} \end{bmatrix} \times \begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

After calculation, we obtain the 6 joint angles of the robot, which are summarized as follows:

$$\theta_1 = \text{atan2}\,(W_xW', W_xO) = \text{atan2}(W_y, W_x) \quad (14)$$

$$\theta_2 = \text{atan2}\left(W_z - d_1, \sqrt{{W_x}^2 + {W_y}^2} - a_1\right) - \text{atan2}(\sqrt{1 - \cos^2\varphi}, \cos\varphi) \quad (15)$$

$$\text{Or } \theta_2 = \text{atan2}\left(W_z - d_1, \sqrt{{W_x}^2 + {W_y}^2} - a_1\right) - \text{atan2}(-\sqrt{1 - \cos^2\varphi}, \cos\varphi) \quad (16)$$

$$\theta_3 = \text{atan2}\left(\sqrt{1 - \cos^2\gamma}, \cos\gamma\right) + \text{atan2}(d_4, a_3) \quad (17)$$

$$\text{Or } \theta_3 = \text{atan2}\left(-\sqrt{1 - \cos^2\gamma}, \cos\gamma\right) + \text{atan2}(d_4, a_3) \quad (18)$$

$$\theta_4 = \text{atan2}(r_{23}, r_{13}) \quad (19)$$

$$\theta_5 = \text{atan2}(\sqrt{1 - {r_{33}}^2}, r_{33}) \quad (20)$$

$$\text{Or } \theta_5 = \text{atan2}(-\sqrt{1 - {r_{33}}^2}, r_{33}) \quad (21)$$

$$\theta_6 = \text{atan2}(r_{32}, -r_{31}) \quad (22)$$

3. Control System Design

The system uses a 220 V supply, converted by a switching power supply to 24 V DC at 40 A to power the system. An LM2596 buck converter then steps this down to 5 V DC for the control board. The STM32F407VET6 microcontroller manages the robot's operation by communicating with the driver controllers and receiving signals from proximity sensors to determine the home position. These sensors are connected through an isolation circuit to ensure electrical separation and protect the control circuitry. A cooling fan is installed for heat diss dissipation.

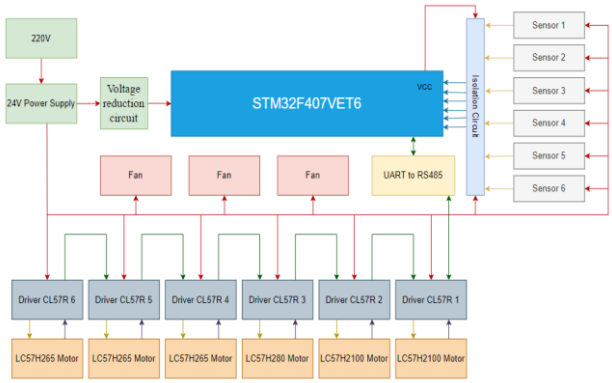

Figure 5. *Diagram showing the components of the control system*

4. Completed system model

The robotic arm is designed with the following technical specifications: it operates on a standard input voltage of 220V and has a total power consumption of 720W. The arm features an effective workspace radius of approximately 802.9 mm, allowing it to reach and manipulate objects within a broad operational range. It offers high object detection accuracy at around 99%, ensuring reliable recognition of items during operation. Additionally, the robotic arm is capable of lifting objects with a maximum load capacity of about 2 kilograms, making it suitable for handling lightweight components in industrial and automation environments.

Table 6. *Detailed specification of the joints of the robot arm*

Joint	Angle limit (Degrees)	Maximum speed (Degrees/second)	Torque (Nm)
O_1	$-180 \div 180$	111	162
O_2	$-20 \div 180$	37.5	480
O_3	$-105 \div 90$	60	220
O_4	$-180 \div 180$	600	30
O_5	$-180 \div 180$	200	90
O_6	$-180 \div 180$	6000	1.25

Electrical system is shown in Figure 6.

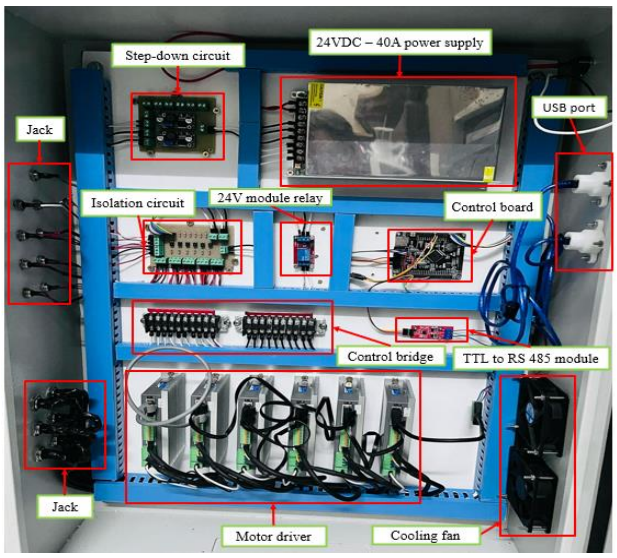

Figure 6. *Electrical cabinet system*

Overall model of the system is shown in Figure 7.

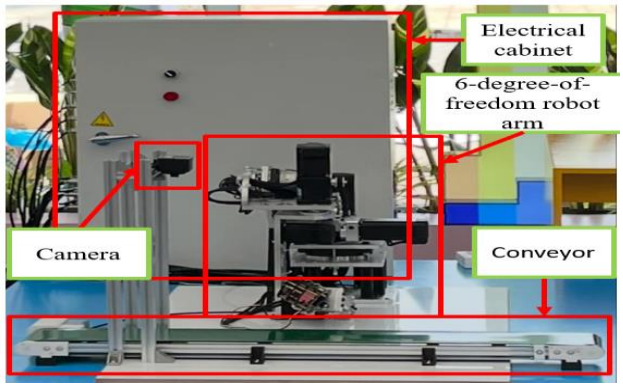

Figure 7. *Overall model of the actual system*

Position error is the distance between the desired position and the actual position of the arm:

This error is calculated as below:

$$\text{Error} = \sqrt{\begin{matrix}(\text{Px}_{\text{desired}} - \text{Px}_{\text{actual}})^2 \\ + (\text{Py}_{\text{desired}} - \text{Py}_{\text{actual}})^2 \\ + (\text{Pz}_{\text{desired}} - \text{Pz}_{\text{actual}})^2\end{matrix}} \quad (23)$$

Average error is calculated as

$$\text{Average error} = \frac{1}{n}\sum_{i=1}^{n} \text{Error}_i \quad (24)$$

where n is the number of trials

The arm moves to the target position and records its actual position; the position error is computed using Equation (23) over multiple trials to build a dataset, and the mean error across all trials is then calculated via Equation (24):

- The average position error of the robot arm during movement: 0.85036 mm (The error is within the initially set range).

- The average position error when determining the position with the camera: 1.37 mm.

5. Image processing

5.1. Calculation formulas

5.1.1. Centroid formula

The centroid is calculated based on the average coordinates of all the pixels belonging to the object. The formula to determine the centroid is as follow:

$$centroid_x = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (25)$$

$$centroid_y = \frac{1}{n}\sum_{i=1}^{n} y_i \quad (26)$$

Where: n is the number of pixels belonging to the object, and x_k, y_k are the coordinates of each pixel.

5.1.2. Size calculation formula

Determine the length and width of the object by computing the coordinated distance between the object's pixels.

$$\text{Length} = \sqrt{(X1 - X2)^2 + (Y1 - Y2)^2} \quad (27)$$

5.1.3. Accuracy determination formula

Convention: Overall accuracy means the average accuracy of the three types of boxes. Values are calculated

by multiplying by 10 and dividing by 12.5 to convert from camera measurements (pixels) to actual size (mm).

Determine centroid accuracy – $X, Y_{Centroid\ 4}$ is the centroid calculated after determining the 4 points of the bounding box, while $X, Y_{Centroid}$ is the centroid determined using formulas (25) and (26). Formula (28) is calculated in millimeters, whereas formulas (29), (30), and (31) are calculated as percentages.

$$\text{error} = \frac{\sqrt{(X_{\text{Centroid 4}} - X_{\text{Centroid}})^2 + (Y_{\text{Centroid 4}} - Y_{\text{Centroid}})^2} * 10}{12{,}5} \quad (28)$$

$$\text{Centroid accuracy} = (1 - \frac{\text{Centroid error}}{10}) * 100\% \quad (29)$$

$$\text{Average centroid accuracy} = \frac{1}{n}\sum_{i=1}^{n} \text{Centroid accuracy}_i \quad (30)$$

(n is the number of trials)

$$\text{Overall centroid accuracy} = \frac{1}{k}\sum_{i=1}^{k} \text{Average centroid accuracy}_i \quad (31)$$

Determine size accuracy, length is the dimension determined by the camera. Formula (32) and (33) are calculated in millimeters, whereas formulas (34), (35), (37), and (38) are calculated as percentages. [5, 6, 7]

$$\text{Length error} = \left|\frac{\text{Length} * 10}{12{,}5} - \text{Actual length}\right| \quad (32)$$

$$\text{Actual length} = \begin{cases} 77\text{mm (If it is a single} - \text{compartment box)} \\ 152\text{mm (If it is a two} - \text{compartment box)} \\ 227\text{mm (If it is a three} - \text{compartment box)} \end{cases}$$

$$\text{Width error} = \left|\frac{\text{Width} * 10}{12{,}5} - \text{Actual width}\right| \quad (33)$$

$$\text{Actual width} = \begin{cases} 76\text{mm (If it is a single} - \text{compartment box)} \\ 76\text{mm (If it is a two} - \text{compartment box)} \\ 81\text{mm (If it is a three} - \text{compartment box)} \end{cases}$$

$$\text{Length accuracy} = [1 - \frac{\text{Length error}}{\text{Actual length}}] * 100\% \quad (34)$$

$$\text{Width accuracy} = [1 - \frac{\text{Width error}}{\text{Actual width}}] * 100\% \quad (35)$$

$$\text{Size accuracy} = \text{Length accuracy} * \text{Width accuracy} \quad (36)$$

$$\text{Average size accuracy} = \frac{1}{n}\sum_{i=1}^{n} \text{Size accuracy}_i \quad (37)$$

$$\text{Overall size accuracy} = \frac{1}{k}\sum_{i=1}^{k} \text{Average size accuracy}_i \quad (38)$$

Determining box type recognition accuracy: If the system correctly identifies the box type, the recognition accuracy is 100%; otherwise, it is 0%. Formulas (39), and (40) are calculated as percentages.

$$\text{Average box accuracy} = \frac{1}{n}\sum_{i=1}^{n} \text{Box accuracy}_i \quad (39)$$

$$\text{Overall box accuracy} = \frac{1}{k}\sum_{i=1}^{k} \text{Average box accuracy}_i \quad (40)$$

Average center accuracy is denoted as ACA, average size accuracy as ASA, and average box type recognition accuracy as ABTRA

$$\text{Accuracy}_i = \text{ACA}_i * \text{ASA}_i * \text{ABTRA}_i \quad (41)$$

$$\text{Overall accuracy} = \frac{1}{k}\sum_{i=1}^{k} \text{Accuracy}_i \quad (42)$$

5.2. Experimental results

In image processing, the accuracy of the box's center, size, and type is evaluated for 1-, 2-, and 3-compartment boxes, and then overall averages are calculated.

- Center accuracy is computed using Equations (30) and (31) by comparing the detected centroid with the ground truth.

- Size accuracy follows Equations (37) and (38), comparing measured dimensions to actual dimensions.

- Type accuracy uses Equations (39) and (40) to evaluate correct classification of 1-, 2-, and 3-compartment boxes.

- Overall accuracy (Equations 41 and 42) is the average of the three metrics above, reflecting the system's overall performance for each box type.

Table 7. *Box Detection Accuracy*

Box Type	Center Accuracy	Size Accuracy	Type Accuracy	Overall Accuracy
1-compartment	96.35%	94.45%	100%	91.00%
2-compartment	97.81%	93.22%	100%	91.18%
3-compartment	98.03%	92.69%	100%	90.87%
Average	97.40%	93.45%	100%	**91.02%**

Overall accuracy exceeds 90%, with 2-compartment boxes highest at 91.18%, then 1-compartment at 91.00% and 3-compartment at 90.87%, demonstrating excellent image processing performance.

6. Simulation of the robot arm's trajectory when picking up a can from a fixed position

The robotic arm operates as follows. From the initial position, it moves to the can pickup point, grips and lifts the can. Next, it travels straight up above the drop-off location, lowers down beside the box to release the can, then lifts up and returns to the pickup point (if more cans remain). Once all cans have been dropped, the arm stops at the final drop-off position.

Actual pick-up times were 11.6 s for one can, 26.0 s for two cans, and 40.4 s for three cans:

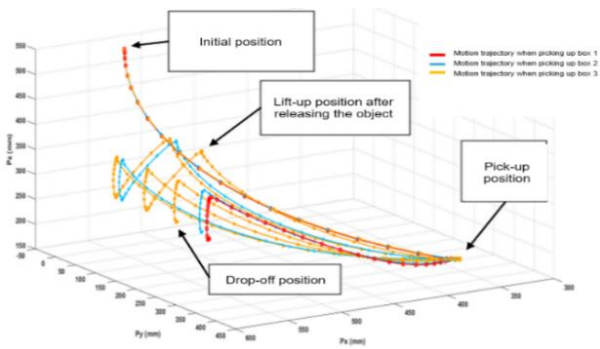

Figure 8. *The motion trajectory of the arm to pick up cans in 3D space is simulated using MATLAB*

7. Training an AI model using YOLO

The AI training process with YOLOv8s-obb [9] is as follows:

- Step 1: Collect and label 2,600 images of cans in various poses and lighting conditions.

- Step 2: Split the dataset into 80% for training and 20% for validation. This ensures that there are enough image examples to train and evaluate the performance of the model.

- Step 3: Run the training command or script, adjusting parameters like batch size, epochs, and input resolution.

After training the YOLO model, we obtained the accuracy in detecting the orientations of the cans as follows:

Table 8. *Summary table of accuracy for each can orientation and overall accuracy*

Category	Precision	Recall	mAP50	mAP50-95
Overall	99.5%	99.5%	99.4%	99.4%
Can Bottom	99.3%	100%	99.5%	99.5%
Upright Can	99.4%	99.7%	99.5%	99.3%
Upside-Down Can	99.4%	99.0%	99.3%	99.2%
Can Lid	99.9%	99.5%	99.5%	99.5%

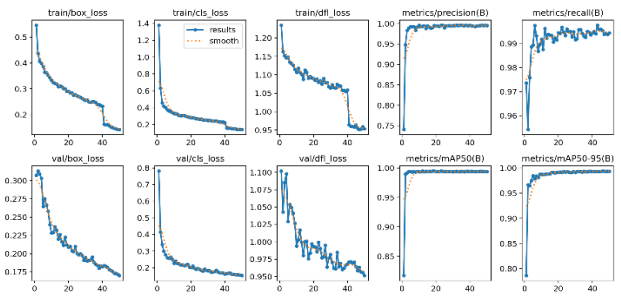

Figure 9. *Training and Validation Metrics over Epochs*

The YOLO model excels in detecting can orientations, achieving 99.5% Precision and Recall, with 99.4% for both mAP@50 and mAP@50-95. Can lids show the highest precision (99.9%) and can bottoms have perfect recall (100%). Upright and upside-down cans maintain high accuracy (99.0%–99.4%), confirming the model's robustness for high-precision detection in complex scenarios.

8. Conclusion

The study successfully developed a modular 6-DOF robotic arm that flexibly meets design specifications and, via image processing, accurately detects cargo box centers, sizes, and types as well as can orientations. Position errors between camera detection and arm movement remained low, satisfying initial requirements.

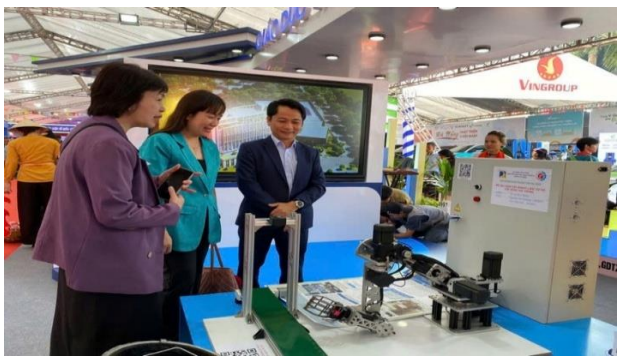

Figure 10. *The robotic arm performance at the "50th Anniversary Da Nang - Development and Integration" exhibition, 2025*

Experimental results demonstrate system feasibility despite limitations in detection accuracy under varying lighting, pick-up speed, and control flexibility. Future work will expand the training dataset, optimize motion planning for faster picking, and integrate advanced control algorithms for smoother, more precise operation.

REFERENCES

[1] H. N. Le, Q. V. Vo, H. P. V. Ngo, N. T. Vo, and A. D. Pham, "Behavior analysis of soft pneumatic actuator gripper by using image processing technology", in *Proc. IEEE Int. Conf. Mechatronics Autom. (ICMA)*, Japan, 2020, pp. 953-958.

[2] V. N. Thanh, D. P. Vinh, L. H. Nam, N. T. Nghi, and D. Le Anh, "Reinforcement Q-learning PID controller for a restaurant mobile robot with double line-sensors", in *Proc. 4th Int. Conf. Mach. Learn. Soft Comput.*, Viet Nam, 2020, pp. 237-241.

[3] V. N. Thanh and H. Sawada, "Simplified cerebellum-like spiking neural network as short-range timing function for the talking robot", *Connection Science*, vol. 30, no. 4, pp. 388-408, 2018.

[4] V. N. Thanh and H. Sawada, "Automatic vowel sequence reproduction for a talking robot based on PARCOR coefficient template matching", *IEIE Trans. Smart Process. Comput.*, vol. 5, no. 3, pp. 215-221, 2016.

[5] H. Sawada and T. V. Nhu, "Intoning speech performance of the talking robot for vietnamese language case", in *Proc. Int. Symp. Micro-NanoMechatronics Human Sci. (MHS)*, Japan, 2018, pp. 1-6.

[6] H. Sawada and T. V. Nhu, "Singing performance of the talking robot with newly redesigned artificial vocal cords", in *Proc. IEEE Int. Conf. Mechatronics Autom. (ICMA)*, China, 2017, pp. 1628-1633.

[7] T. H. T. Tran, D. S. Nguyen, N. T. Vo and H. N. Le, "Design of Delta Robot Arm based on Topology optimization and Generative Design Method", in *Proc. 5th Int. Conf. Green Technol. Sustainable Dev. (GTSD)*, Ho Chi Minh City, Vietnam, 2020, pp. 157-161.

[8] H. N. Le and N. T. Vo, "Modeling and analysis of an RUU Delta Robot using SolidWorks and SimMechanics", *Int. J. Dynamics Control*, vol. 12, no. 7, pp. 2467-2479, 2024.

[9] N. T. Vo, C. T. Pham, V. T. Vo, V. T. Ho, M. Q. Tran, and N. N. M. Hoang, "Traffic sign recognition through the use of an Internet of Things system and deep learning", *Inf. Control Syst.*, vol. 135, no. 2, pp. 2-15, 2025.

[10] A.-D. Pham, B. H. Thai, P. V. Dang, and N. T. Vo, "Analysis of the parametric configuration impact on BallBot control performance", *Int. J. Mech. Syst. Dyn.*, vol. 4, no. 4, pp. 446-460, 2024.

[11] L. Patrício, L. Varela, and Z. Silveira, "Integration of Artificial Intelligence and Robotic Process Automation: Literature Review and Proposal for a Sustainable Model", *Applied Science.*, vol. 14, pp. 9648, 2024.

[12] N. T. Vo, Q. N. Dinh, D. M. T. Nguyen, and A. D. Pham, "Development of an automatic egg fertility detection system applying image processing technique", *The University of Danang - Journal of Science and Technology*, vol. 22, no. 9A, pp. 7–11, 2024.

[13] A. Borboni et al., "The Expanding Role of Artificial Intelligence in Collaborative Robots for Industrial Applications: A Systematic Review of Recent Works", *Machines*, vol. 11, no.1, ID. 111, 2023.

[14] R. K. Megalingam et al., "Integration of Vision based Robot Manipulation using ROS for Assistive Applications", in *Proc. Second Int. Conf. Inventive Research Comput. Appl. (ICIRCA)*, India, 2020, pp. 163-169.

[15] D. Marrero, J. Kern, and C. Urrea, "A Novel Robotic Controller Using Neural Engineering Framework-Based Spiking Neural Networks", *Sensors*, vol. 24, ID. 491, 2024.

[16] V. N. Thanh, D. P. Vinh, N. V. Tho, T. Q. Duc, and N. L. Minh, "Application of bluetooth mesh network in multi-device system management", *The University of Danang - Journal of Science and Technology,* vol. 22, no.3, pp. 7–12, 2024.

[17] A. Nussibaliyeva et al., "Development of an Artificial Vision for a Parallel Manipulator Using Machine-to-Machine Technologies", *Sensors*, vol. 24, ID. 3792, 2024.

[18] V. N. Thanh, "Development of restaurant serving robot using line following approach", *The University of Danang - Journal of Science and Technology,* vol. 17, no. 12.1, pp. 1-4, 2020.

[19] Y. Park, J. Lee, D. Sim, Y. Cho, and C. Park, "Designing Spiking Neural Network-Based Reinforcement Learning for 3D Robotic Arm Applications", *Electronics*, vol. 14, ID. 578, 2025.