The Development of Robotic Manipulator for Automated Test Tube

Thi Thoa Mac[1], Dam Nhan Trinh[1], Van Thanh Tri Nguyen[1], Thanh-Hung Nguyen[1], Levente Kovács[2], Tien-Duc Nguyen[1*]

[1]Department of Mechatronics, School of Mechanical Engineering, Hanoi University of Science and Technology, Hanoi, Vietnam; thoa.macthi@hust.edu.vn; nhan.td195802@sis.hust.edu.vn; tri.nvt200634@sis.hust.edu.vn; hung.nguyenthanh@hust.edu.vn

[2]Physiological Controls Research Center, University Research and Innovation Center, Óbuda University, Budapest, Hungary, kovacs@uni-obuda.hu

[*]Corresponding author: Tien-Duc Nguyen (duc.nt230014d@sis.hust.edu.vn)

Abstract: In recent years, robotics has experienced significant development and widespread application across various manufacturing sectors. This progress has been driven by the integration of breakthroughs in technology such as artificial intelligence and computer vision, enabling robots to become more intelligent and adaptable when performing specific tasks. As a result, demand for integrating robots into human production and research activities has been accelerated. Specifically, in chemical-related industries, reducing or avoiding direct human contact with chemicals is essential for ensuring the safety of performers. In laboratory settings, automating tasks such as the arrangement of chemical tubes using robotic arms has emerged as a solution to enhance safety and save time for researchers. Building upon this concept, this paper presents a robotic system that serves as a laboratory assistant for arranging centrifuge tubes into trays. The system is composed of a 5-degree-of-freedom robot arm, Reactor X-150, alongside depth camera D435 and computer vision model YOLOv8. By collecting image recognition information from YOLOv8 and analyzing it in conjunction with depth camera data, the system determines the position and orientation of the tube, which is then transmitted to the robot for the arrangement process. This integrated approach aims to improve safety in handling centrifugation experiments.

Keywords: Arm manipulator; automation; centrifuge tube

1 Introduction

Over the past few years, there has been significant growth in the domain of Industry 4.0 [1]. This progress can be attributed to notable advancements in technology, resulting in increased sophistication and ease of implementation.

Therefore, there has been a surge in popularity and broader application of robots across diverse industries and fields, with an increasing focus on tackling progressively complex tasks and responsibilities [2].

The integration of robots in both production and research stages has proven to be instrumental in automating processes, thereby either replacing or assisting human workers in the production process [3]. The presence of robots represents a significant advancement that has had a profound impact on a wide range of industries and fields. By using the advanced capabilities of robots, previously human-performed tasks can now be executed with increased precision and efficiency. In the research domain, robots play a crucial role in automating tasks such as data collection, analysis, and experimentation, resulting in accelerated research timelines and improved reliability of outcomes. The collaborative interaction between humans and robots in these stages not only leads to improved operational efficiency but also allows human workers to focus on more complex and value-added aspects of their work [4].

In the chemical and medical industries, it is customary to limit direct exposure to chemicals or specimens because of the fragility and vulnerability of the human body to highly reactive chemicals [5-6]. Moreover, specimens often carry highly infectious agents that can lead to cross-transmission, exacerbating the severity of diseases. It is crucial to note that even a minor error in a worker's performance can result in inaccurate results and, in the worst-case scenario, cause irreversible harm to their health. Therefore, one effective approach to addressing these weaknesses is the utilization of robots to assist in various aspects of the work process [7].

The integration of artificial intelligence (AI) and computer vision technologies has revolutionized the capabilities of robots in the chemical and medical industries. By incorporating machine learning techniques, robots have gained a heightened level of flexibility and adaptability [8]. Through AI and computer vision, high-quality sensors, the Internet of Things (IoT), and cloud computing, robots can analyze and understand data from their environment, enabling them to make informed decisions and perform assigned tasks. This integration enhances the robots' ambient awareness, allowing them to recognize objects, assess spatial relationships, and identify potential hazards. Moreover, the continuous learning capabilities of AI algorithms enable robots to improve their performance over time, optimizing their decision making processes and minimizing errors. Ultimately, the integration of AI and computer vision empowers robots to operate autonomously and efficiently, reducing the risks associated with direct human exposure to dangerous chemicals and infectious specimens while simultaneously improving productivity and safety in these industries [5-6].

This study showcases a system that combines mechanical mechanisms, artificial intelligence, and a depth camera. The Reactor X-150 robotic arm, named Reactor X-150 and manufactured by Trossen Robotics, has been designed and engineered

to ensure smooth control through the utilization of ROS. The artificial intelligence model employed in the system is YOLOv8, a technology developed by Ultralytics that proficiently carries out the task of recognizing test tubes. Additionally, the position and orientation of objects are accurately determined by processing data received from the Intel D435 depth camera.

2 System Setup and Kinematics Analysis

2.1 System Setup

The system prioritizes compactness and minimizes interference caused by lighting. It includes a 400×400×10 millimeter-thick black wooden panel with the robot arm fixed in the corner. The camera, mounted on an aluminum stand, is positioned at a distance of 550 millimeters from the center of the camera to the system table. Adjacent to the left side of the system, a test tube tray is positioned, capable of accommodating up to 24 test tubes.

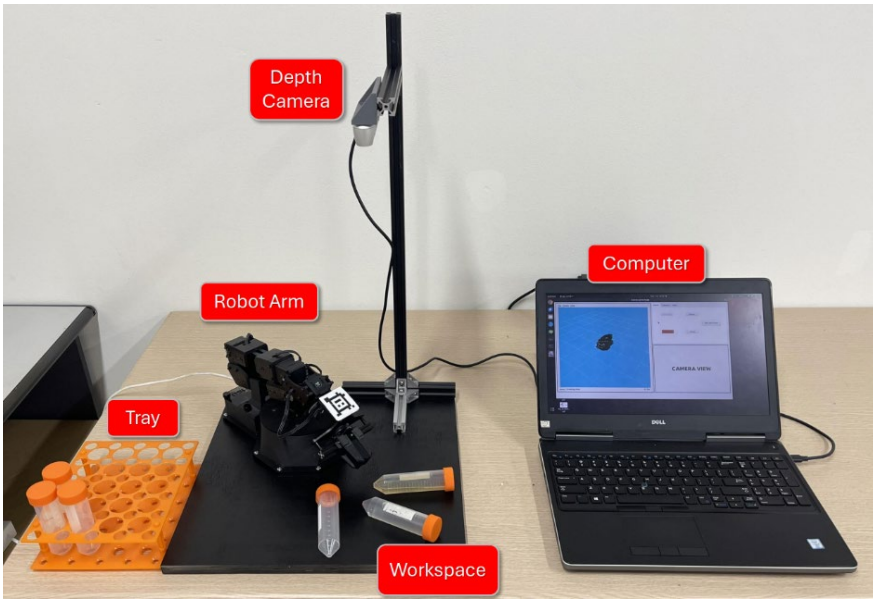


Figure 1
The setup of the system

2.2 Kinematics Analysis

In order to tackle the challenge of solving the forward kinetics problem for the robotic arm, the decision was made to employ the Product of Exponentials (POE) method [10] due to its inherent advantages in terms of user-friendliness and rapid programming. The utilization of POE parameters provides a more simplified and intuitive representation of the robot's kinematics when compared to the conventional Denavit-Hartenberg (DH) parameters, thus facilitating improved comprehension and visualization of the robot's configuration. Additionally, POE parameters offer the advantage of being free from coordinate singularities, thereby enabling greater control and avoiding the loss of degrees of freedom. Moreover, the adoption of POE parameters allows for more efficient and continuous computations for both forward and inverse kinematics, resulting in improved computational efficiency.

Assigning the coordinate system to the structure of the robot, as shown in Fig. 2, and ascertaining the pose of the gripper in relation to the base frame of the robot is a comparatively straightforward task.

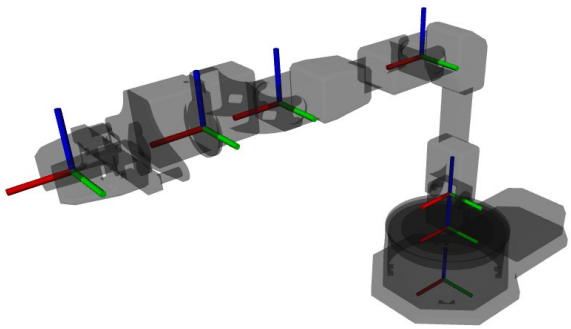


Figure 2
Robot coordinate system

By analyzing the dimensions of the robot structure, the matrix for the default configuration is determined in meters.

$$M = \begin{bmatrix} 1 & 0 & 0 & 0.358575 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0.25457 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

In the POE method, the twist of a revolute joint is composed of two distinct components, namely the translational component $\vec{v}$ and the rotational component $\vec{\omega}$.

The rotational motion $\vec{\omega}$ is a 3×1 vector and can be reorganized in cross product matrix notation as follows:

$$\vec{\omega} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{2}$$

Applying Rodrigues' Rotation Formula, the rotation matrix constructed from the rotation component of a twist is as follows:

$$e^{\vec{\omega}\theta} = I + \vec{\omega}\sin\theta + \vec{\omega}^2(1-\cos\theta) \tag{3}$$

The translation vector calculated from twist components is as follows:

$$t = \left(I - e^{\vec{\omega}\theta}\right)(\vec{\omega} \times \vec{v}) + \omega\omega^T v\theta \tag{4}$$

From the above rotation and translation matrices, the matrix exponential for the i-th joint can be composed as follows:

$$e^{\vec{\xi}_i\theta_i} = \begin{bmatrix} e^{\vec{\omega}\theta} & t \\ 0 & 1 \end{bmatrix} \tag{5}$$

The representation of the twist for the system joints is encapsulated within the S_{list} matrix, which is defined as follows:

$$S_{list} = [\vec{\omega}_i \quad \vec{v_i}]^T = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -0.10457 & 0 & 0 \\ 0 & 1 & 0 & -0.25457 & 0 & 0.05 \\ 0 & 1 & 0 & -0.25457 & 0 & 0.2 \\ 1 & 0 & 0 & 0 & 0.25457 & 0 \end{bmatrix}^T \tag{6}$$

The determination of the relative location of the tool frame in relation to the base frame is accomplished through the following:

$$T_0^{ee} = e^{\vec{\xi}_1\theta_1} e^{\vec{\xi}_2\theta_2} e^{\vec{\xi}_3\theta_3} e^{\vec{\xi}_4\theta_4} e^{\vec{\xi}_5\theta_5} M \tag{7}$$

For solving the system inverse kinematics, we chose to use the Newton-Raphson Approximation Method. The root-finding algorithm produces successively better approximations to the roots of a real-valued function. With an initial guess x_0, the level-1 Taylor expansion of the function is:

$$f(x) = f(x_0) + \frac{f(x_0)}{1!}(x - x_0) \tag{8}$$

Solving this equation $f(x) = 0$, given the initial root x_0:

$$f(x_0)+\frac{f(x_0)}{1!}(x-x_0)=0$$
$$x-x_0=\frac{-f(x_0)}{f'(x_0)} \quad (9)$$
$$x=x_0-\frac{-f(x_0)}{f'(x_0)}$$

The root x and x_1 obtained then becomes the initial root for the next solving iteration until x_k:

$$x_2=x_1-\frac{f(x_1)}{f'(x_1)} \quad (10)$$

Then, the general root formula for every iteration is:

$$x_{k+1}=x_k-\frac{f(x_k)}{f'(x_k)} \quad (11)$$

The solving iteration loop exits if and only if the value x_k meets certain qualifications. Such qualifications are the condition of the initially given error ϵ or the condition of the maximum allowable iterations. The absolute error is determined as:

$$|x_k-x_{k+1}|\leq\epsilon \quad (12)$$

3 Tube Detection Process

3.1 Identify the Appearance of Tubes

YOLOv8, Fig. 3, is the version in the YOLO (You Only Look Once) model series, building upon of YOLOv5 architecture [9]. The prime aim of YOLOv8 is to achieve superior speed, accuracy, and usability, thereby enhancing its responsiveness across various tasks. Several significant changes have been implemented in YOLOv8, including:

- YOLOv8 uses an anchor-free detection strategy by directly predicting object centers, eliminating the need for predefined anchor boxes.

- YOLOv8 replaces the initial 6×6 convolutional layer with a more effective 3×3 convolutional layer, leading to improved feature extraction capabilities.
- YOLOv8 introduces the C2f module to effectively merge high-level features with contextual information, enhancing object detection accuracy in complex scenes.
- YOLOv8 presents a semantic segmentation model referred to as YOLOv8-Seg. This model employs CSPDarknet53 as its core feature extractor and integrates the C2f module. It incorporates two segmentation heads to generate semantic segmentation masks, making it a versatile tool for a range of computer vision applications.

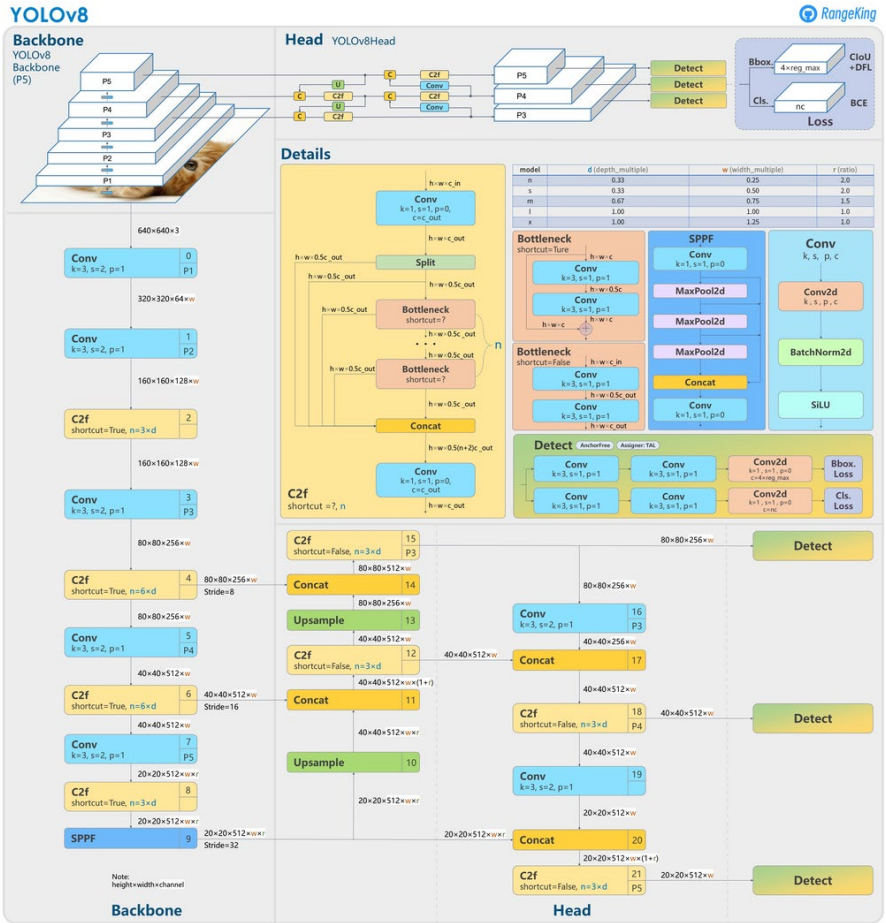


Figure 3
YOLOv8 Architecture

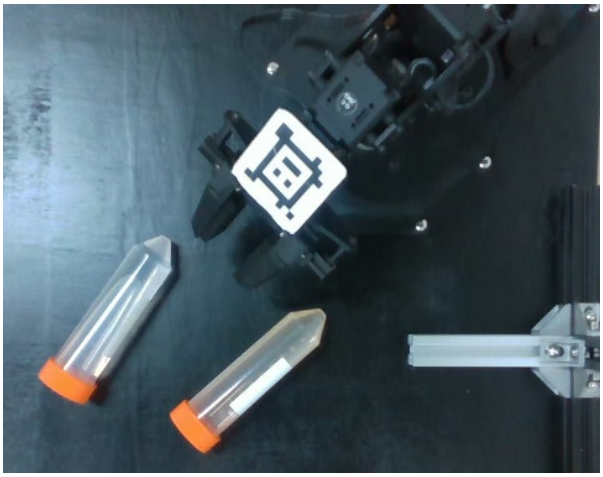


Figure 4
Input image from the depth camera D435

The integration of the new Ultralytics model, YOLOv8, into the system has demonstrated a remarkably straightforward process. After being trained with custom data tailored to centrifuge tubes, the model uses segmentation to a produce preliminary result (Fig. 5) from the input image (Fig. 4).

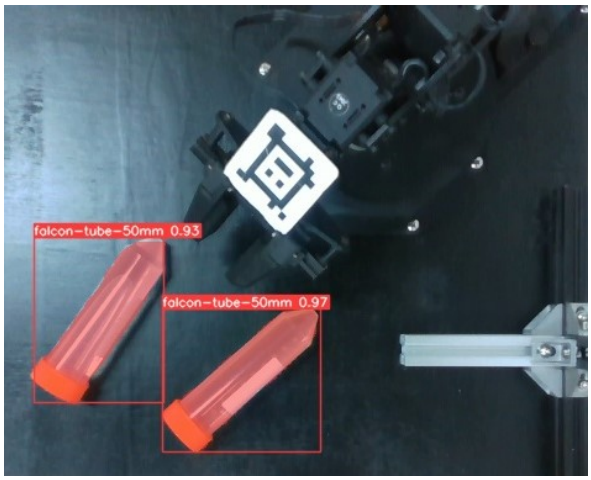


Figure 5
Output from the YOLOv8 model

However, it is of utmost importance to recognize that these results are provisional, as it is crucial to verify whether the test tubes have been appropriately capped. Transporting uncapped test tubes may lead to liquid spillage, presenting potential

hazards to individuals. With the primary objective being the handling of exclusively capped test tubes, it is recommended to implement a capping procedure to verify the status of each individual test tube.

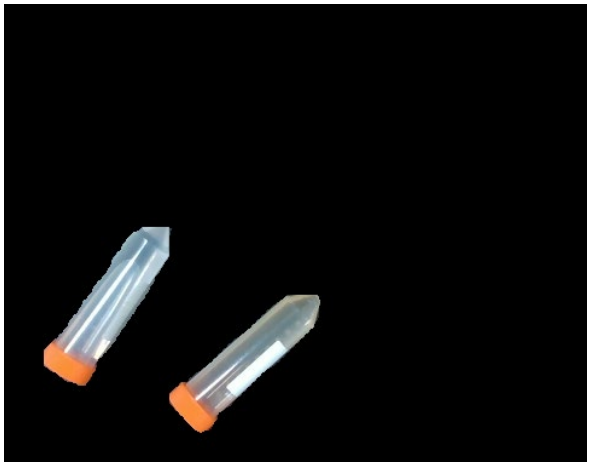


Figure 6
Extracting objects field from output of YOLOv8 model

To carry out the task, the artificial intelligence model's output will consist of image regions containing objects against a black background, as shown in Fig. 6. Yolo's outcomes provide two essential elements: object contours and Regions of Interest (ROI). By identifying orange areas in each ROI, the caps are detected. Thus, the fulfillment of the object's condition requires two factors: YOLO's recognition of an object as a test tube and the presence of the cap within the same ROI.

3.2 Estimate the Position and Orientation

Once the outlines of the object's body and lid have been obtained, the subsequent step involves determining the centroid for each contour in the pixel frame of reference of the image. In this process, the moment() method in OpenCV [11] is used to calculate the image moment for a specific contour. An image moment represents a weighted average of pixel intensities within an image and enables the extraction of specific properties such as the centroid, area, and orientation. Subsequently, the centroid of the contour in pixel coordinates is displayed below.

$$\begin{cases} C_x = M_{10} / M_{00} \\ C_y = M_{01} / M_{00} \end{cases} \tag{13}$$

The positions of the lid and body centroids are denoted by green and red dots, as shown in Fig. 7.

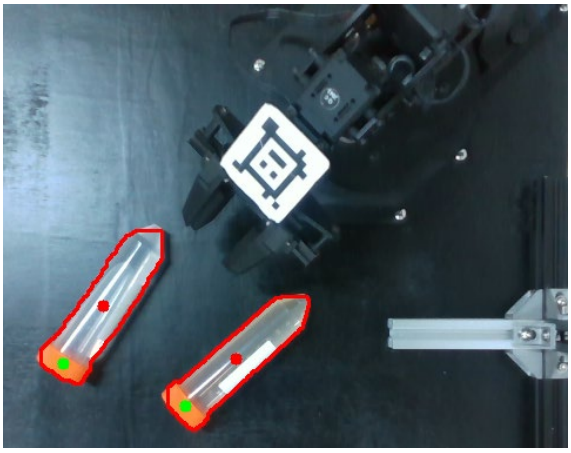


Figure 7
Centroids of centrifuge tubes' lid and body

Before proceeding, it is important to calibrate the camera in order to obtain the Extrinsic and Intrinsic matrices. The Extrinsic matrix represents a transformation matrix that maps the reference coordinate system to the camera coordinate system. On the other hand, the Intrinsic matrix facilitates the transformation between the camera's 3D coordinates and the corresponding 2D image coordinates. Calibrating the camera allows for accurate mapping between the real-world and image space, enabling precise measurements and analysis.

The Intrinsic matrix of camera can be calculated by:

$$InstrinsicMatrix = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 0 \end{bmatrix} \tag{14}$$

Where:

- f_x, f_y: the focal lengths. [pixels]
- c_x, c_y: the image center point. [pixels]

And, these distortion coefficients account for any potential lens distortion are present in the matrix:

$$DistortionCoefficients = [k_1 \quad k_2 \quad t_1 \quad t_2 \quad t_3] \tag{15}$$

The Intel RealSense D435 camera, having undergone calibration, provides an Intrinsic matrix that can be extracted through a ROS topic. These Intrinsic matrix and distortion coefficients, denoted as K and D respectively, encompasses the internal parameters of the camera. It facilitates the transformation between the camera's 3D coordinates and the corresponding 2D image coordinates.

$$K = \begin{bmatrix} 617.329 & 0 & 430.309 \\ 0 & 617.271 & 242.259 \\ 0 & 0 & 1 \end{bmatrix} \quad (16)$$

$$D = [0 \quad 0 \quad 0 \quad 0 \quad 0] \quad (17)$$

Some basic steps are implemented to determine the coordinates of the centroid within the camera's frame of reference. Firstly, retrieve the point cloud data from the Intel RealSense D435 camera topic. At this point the cloud data contains three-dimensional coordinates (X, Y, and Z) for each point. Use the camera's Intrinsic matrix to convert these point cloud coordinates, originally referenced to the camera coordinate system, into pixel coordinates on the corresponding image. Then, by comparing these pixel coordinates with the previously computed coordinates of the centroid, the precise coordinates of the centroid within the camera's frame of reference can be determined. This process establishes a direct correspondence between the point cloud data and the pixel coordinates, allowing for accurate localization of the centroid within the camera perspective.

However, it is important to note that the depth, or z-coordinate, of the centroid of the test tube body is not required for our purposes. This is because the presence of liquid within the tube and the transparency of the tube material can interfere with or prevent accurate depth measurement using the camera's laser, thereby reducing the accuracy of the depth information obtained. Consequently, our focus is solely on determining the depth of the tube cap. However, it should be noted that the depth measurement obtained corresponds only to the rim of the lid. To obtain the actual depth of the center of the lid, it is necessary to increase this depth by precisely the radius of the lid, which in this case is 16 millimeters. By applying this adjustment, we can determine the true depth of the center of the test tube lid within the camera's frame of reference.

Figure 8
AprilTag's position on the robot

The utilization of AprilTag provides the flexibility to establish the position relationship between the robot and the camera without relying on fixed mechanical fixtures. This allows for movement within the camera's field of view. AprilTag serves as an intermediary coordinate system, bridging the coordinates of

the robot and the camera through the implementation of automatic location and orientation recognition algorithms. In this particular case, the 41H12 Standard Family of AprilTags is employed. The tag is securely affixed to a fixed location, as illustrated in Fig. 8, and is defined as an integral component of the robot within the Unified Robot Description Format (URDF).

Finally, the camera reference frame is converted to the base robot frame, allowing seamless integration of visual information, Fig. 9.

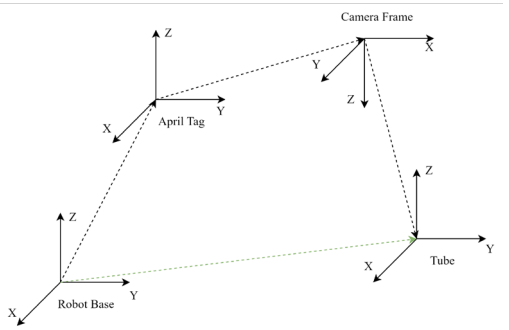


Figure 9
Frame transformation diagram

After collecting and processing data from the camera, the properties of tubes are shown in a virtual environment, Fig. 10. The arrow symbol represents the test tube, with the arrowhead denoting the lid and the arrow tail representing the tube's body and position. This transformation enables accurate perception and interaction between the robot and the test tube, facilitating effective manipulation and control.

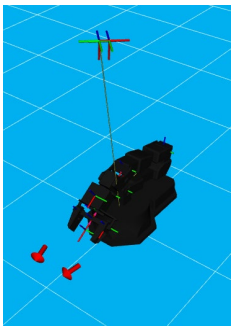


Figure 10
Tubes properties are shown in virtual environment

4 Rotbot Arm Movement Generation

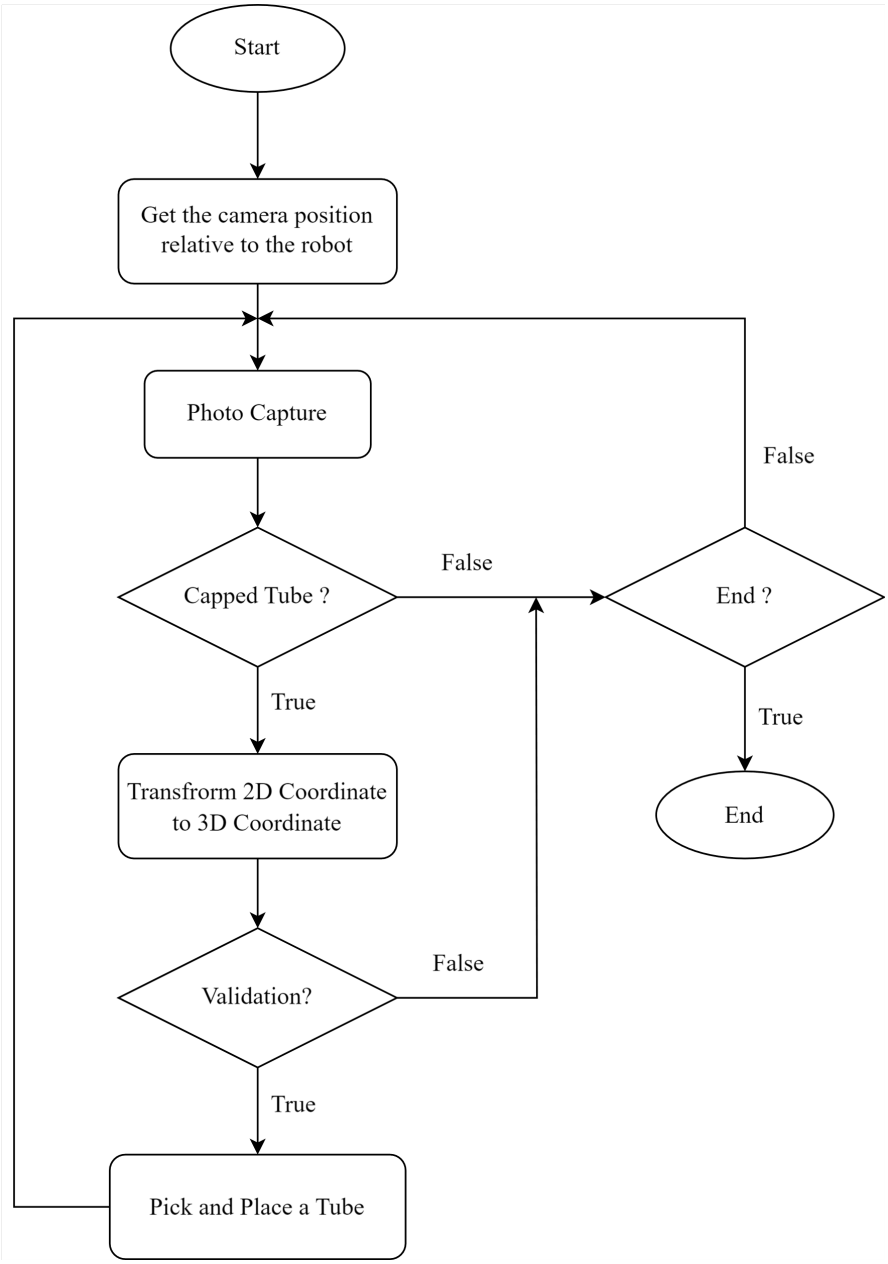


Figure 11
System flow chart

The provided flowchart, Fig. 11, outlines the operation of the system in a formal manner. It begins by receiving information or a matrix that facilitates switching between the robot and the camera. Subsequently, the system enters a loop where it repetitively performs the tasks of recognizing and picking up all the tubes within its capacity. Within each iteration of the loop, the system carries out tube recognition, using appropriate techniques to identify and locate the tubes based on visual input from the camera. Following tube recognition, the system proceeds with the picking-up operation, employing the robot's manipulator or gripper to execute precise movements for grasping and lifting the tubes. This loop continues until all the tubes within the system's capacity have been recognized and picked up, leading to the completion of the system's operation.

The diagram below, Fig. 12, shows the trajectory of the robot in the pick-and-place process of the tube.

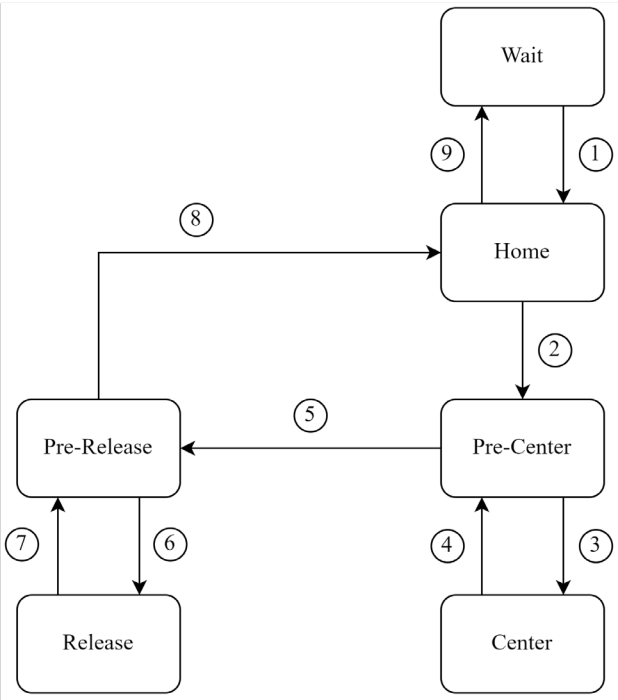


Figure 12
Pick-and-place sequence for each tube diagram

More specifically, when moving to the "pre-center" position, the gripper will rotate at the following angle along its axis:

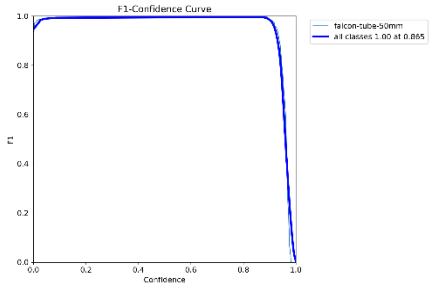
$$\theta_5 = \arctan 2\left(C_y, C_x\right) - \arctan 2\left(L_y - C_y, L_x - C_x\right) \tag{18}$$

Where:

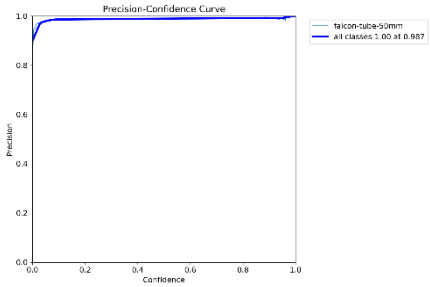
- L_x, L_y: x, y-coordinates of Tube Lid. [m]
- C_x, C_y: x,y-coordinates of Tube Body. [m]

5 Experiment Results

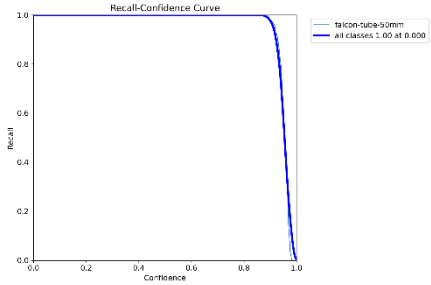
From a dataset with 500 samples, some evaluations of the YOLOv8 model training results are provided. In Fig. 13, the F1 score curve shows the model great confidence since our dataset is composed of 1 class, as the decline threshold is 0.865, while the precision-confidence and recall-confidence curve show the model's excellence in predictions, as both plots have threshold values higher than 0.8.



(a) F1-Confidence Curve



(b) Precision-Confidence Curve



(c) Recall-Confidence Curve

Figure 13
Confidence curves of YOLOv8 after trainning

These plots shows the decrease of loss functions of the training and validating process over 100 epochs, which is composed of bounding box prediction loss, instant segmentation loss and classification loss.

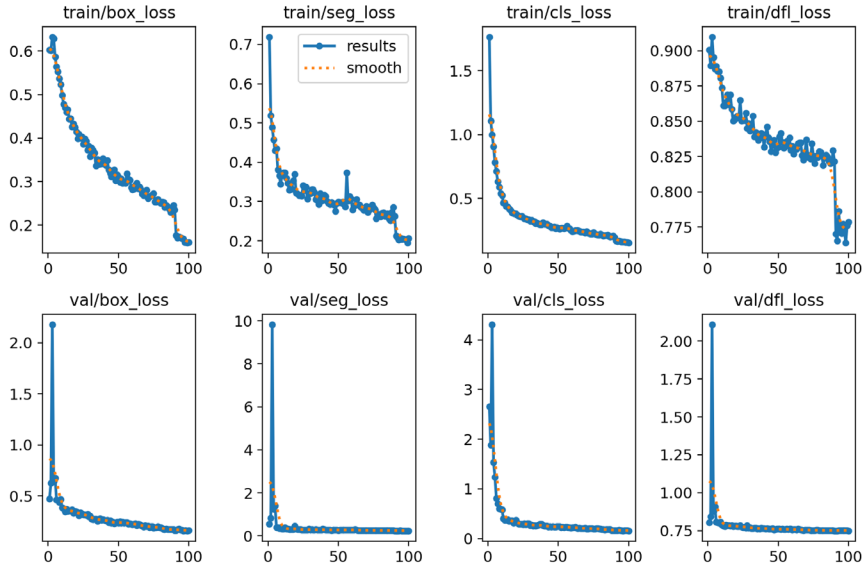


Figure 14
Loss plots in training and validation YOLOv8

Upon implementation in the actual system, we acquired parameters that yielded results as depicted in the following data. Tabe 1 shows the change in position of each joint on different nodes in the pick-and-place trajectory, which is shown in Fig. 15. Additionally, Figs. 16 and 17 describe joint position and velocity over time, respectively. Overall, the system operated in compliance with the stipulated requirement of accurately grasping the test tube from its designated location and subsequently placing it in the designated tray. For each tube, the robot covered a total distance of 2.537 meters during its travel, completing the task within an average time frame of 34.24 seconds. The system tested gave a successful result of 72 out of 100 trials, most of the unsuccessful results were due to missing holes in the tray.

Table 1
Joint Position Sequence

Node	**Joint angle (rad)**				
	Joint 0	**Joint 1**	**Joint 2**	**Joint 3**	**Joint 4**
1	-0.0015	-1.7932	1.5646	0.8022	0.0107
2	-0.0015	-1.7381	1.5217	0.7977	0.0107

3	-0.0015	-1.5412	1.1029	0.5906	0.0107
4	-0.0015	-0.9482	0.8590	0.4648	0.0107
5	-0.0015	-0.8928	0.8130	0.4402	0.0107
…	…	…	…	…	…
386	-0.0015	-1.7932	1.5646	0.8007	0.0107
369	-0.0015	1.7947	1.5631	0.8007	0.0107

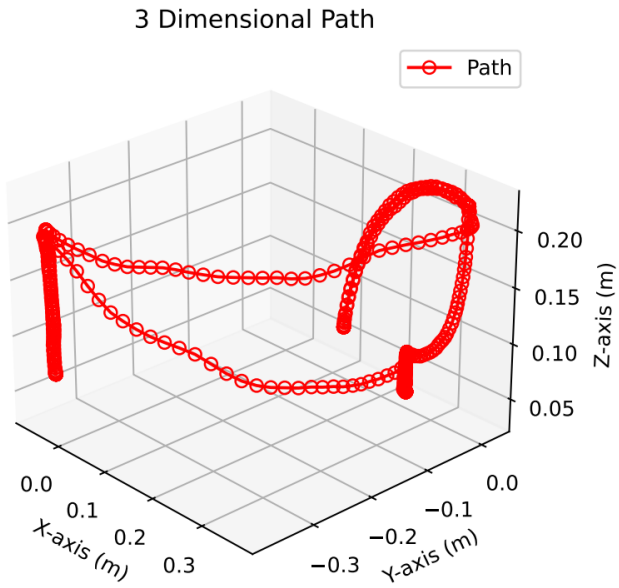


Figure 15
3-Dimensional path

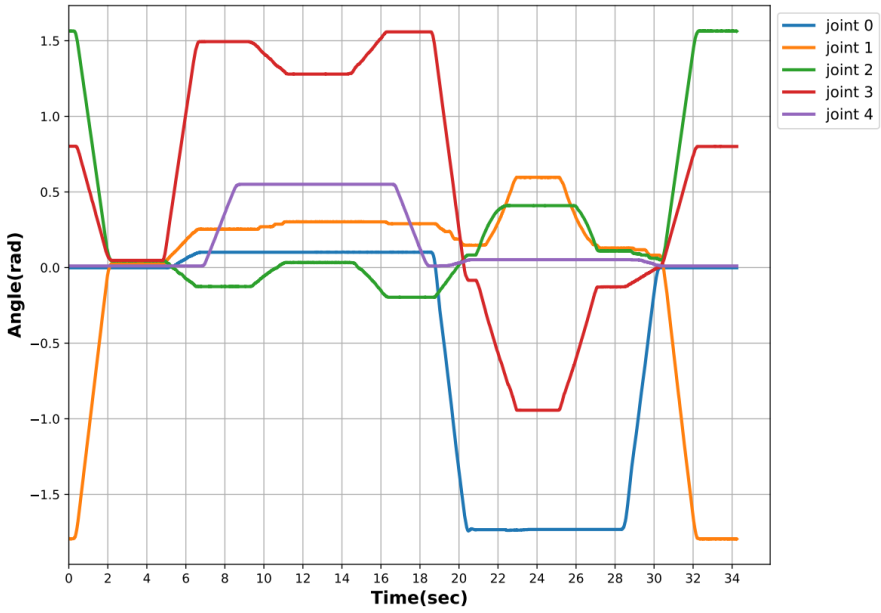


Figure 16
Joints position plot

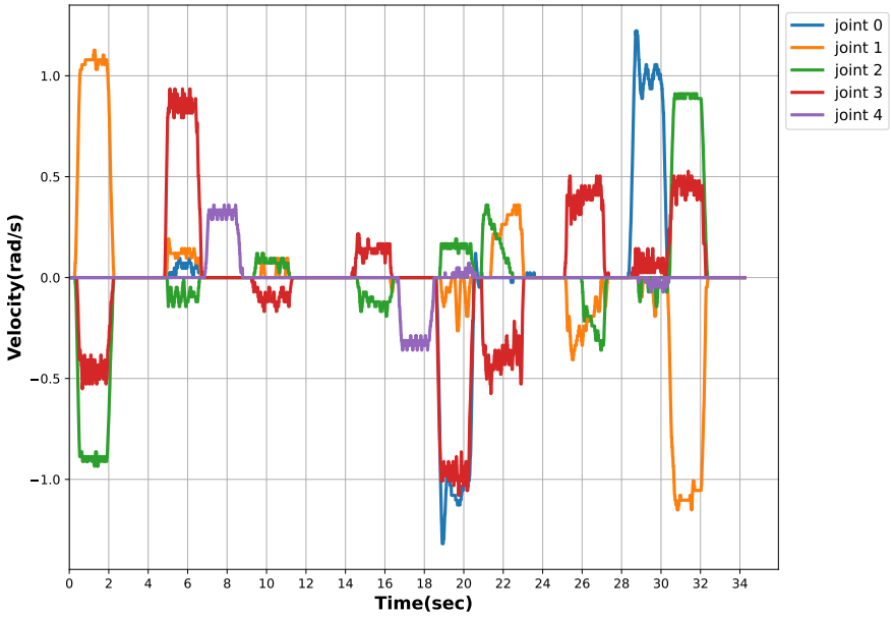


Figure 17
Joints velocity plot

Conclusions

The presented article introduces an intelligent robotic system designed to assist individuals in the production process. By integrating YOLOv8 and computer vision technologies, the robotic arm could pick and place test tubes into the tray effectively. However, it is important to acknowledge that the system is subject to certain assumptions, such as the test tube being parallel to the tabletop and negligible differences between the tray hole and the tube. Moreover, the limit in this paper experiment also lays on the material of the centrifuge tube, which is plastic instead of glass. Another disadvantage of this research is that the peg-in-hole, or putting the test tube down to the hole on the tray, and arranging tubes in the tray require deep research in the controller, so it may be a future task. Consequently, the current solution may not be considered a comprehensive, optimal approach. Nonetheless, the system displays significant potential for further development, particularly in the realm of human-machine interaction. Potential avenues for future enhancements include integrating a pick-up direction predictor for the robot, optimizing the robot's movement path, enhancing the point cloud processing capability for transparent objects, and enabling human interaction detection. These directions offer promising prospects for advancing the functionality and performance of the system.

Acknowledgement

This research is funded by Hanoi University of Science and Technology (HUST) under project number T2022-PC-023.

References

[1] Ray Y. Zhong, Xun Xu, Eberhard Klotz, Stephen T. Newman: Intelligent Manufacturing in the Context of Industry 4.0: A Review, Engineering, Vol. 3, No. 5, 2017, pp. 616-630

[2] Shiyong Wang, Jiafu Wan, Daqiang Zhang, Di Li, Chunhua Zhang: Towards smart factory for industry 4.0: a self-organized multi-agent system with big data based feedback and coordination, Computer Networks, Vol. 101, 2016, pp. 158-168

[3] Loris Roveda, Mauro Magni, Martina Cantoni, Dario Piga, Giuseppe Bucca: Human–robot collaboration in sensorless assembly task learning enhanced by uncertainties adaptation via Bayesian Optimization, Robotics and Autonomous Systems, Vol. 136, 2021, 103711

[4] Loris Roveda: Adaptive interaction controller for compliant robot base applications, IEEE Access, 2018, 7: 6553-6561

[5] Hao Chen, Weiwei Wan, Masaki Matsushita, Takeyuki Kotaka, Kensuke Harada: Robotic Test Tube Rearrangement Using Combined Reinforcement Learning and Motion Planning, arXiv preprint: arXiv:2401.09772, 2024

[6] Mohd Javaid, Abid Haleem: Industry 4.0 applications in medical field: A brief review, Current Medicine Research and Practice, Vol. 9, No. 3, 2019, pp. 102-109

[7] Manogaran, G., Thota, C., Lopez, D., Sundarasekar, R.: Big Data Security Intelligence for Healthcare Industry 4.0. In: Thames, L., Schaefer, D. (eds) Cybersecurity for Industry 4.0. Springer Series in Advanced Manufacturing. Springer, Cham. https://doi.org/10.1007/978-3-319-50660-9_5, 2017

[8] Mohd Javaid, Abid Haleem, Ravi Pratap Singh, Rajiv Suman: Substantial capabilities of robotics in enhancing industry 4.0 implementation, Cognitive Robotics, Vol. 1, 2021, pp. 58-75

[9] Jacob Solawetz, Francesco: What is YOLOv8? The Ultimate Guide. Roboflow Blog, 2023. https://blog.roboflow.com/whats-new-in-YOLOv8/

[10] LYNCH, Kevin M.; PARK, Frank C. Modern robotics. Cambridge University Press, 2017

[11] BRADSKI, Gary. The opencv library. Dr. Dobb's Journal: Software Tools for the Professional Programmer, 2000, 25.11: 120-123