Development of an Intelligence Vision for a Robot System to Pick and Place Objects

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

Phan Xuan Trung
Industrial Maintenance Training Center
Ho Chi Minh City University of Technology (HCMUT), VNU-HCM, Ho Chi Minh City
Vietnam

Phan Trong Quyen
Industrial Maintenance Training Center
Ho Chi Minh City University of Technology (HCMUT), VNU-HCM, Ho Chi Minh City
Vietnam

Tran Thi Truc Mai
Industrial Maintenance Training Center
Ho Chi Minh City University of Technology (HCMUT), VNU-HCM, Ho Chi Minh City
Vietnam

This paper presents an automated pick-and-place robotic system utilizing stereo vision technology for object detection and localization in 3D space. Stereo vision is an optimal choice for short-range industrial applications due to its capability of providing accurate depth measurements at a reasonable cost, outperforming alternatives such as LiDAR or Time-of-Flight (ToF) cameras in similar settings. The proposed system is designed to operate reliably under natural lighting conditions, making it well-suited for deployment in factory production lines. An Intel RealSense D435 camera is employed to capture both RGB and depth images from the environment. Object detection is performed using a YOLOv11-based model, achieving high detection accuracy with a mean average precision (mAP50) of 98.5% across all object classes. The system processes depth information to identify the topmost object, estimates its 3D coordinates with minimal errors (average positional errors below 5.3 mm), and transmits the data to a robotic manipulator for execution of the pick-and-place task. Experimental results demonstrate the system's high precision and reliability in object detection and 3D localization.

Keywords: *Real-Time Object Detection; YOLOv11; Industrial robot; 3D vision; Intel RealSense Camera*

1. INTRODUCTION

In pick-and-place applications using robots, computer vision plays an important role. They help robots to recognize the surrounding environment, and determine the position, size, and orientation of objects to be manipulated. The recognition ability of computer vision helps robots operate more accurately in dynamic environments, improving the level of robot automation [1-4]. Integrating computer vision into robotic systems has helped meet the increasing demands in industrial manufacturing, warehousing, and supply chains. Thanks to feedback from computer vision systems, robots today perform tasks with high speed and accuracy in real time [5-6]. Integrating vision systems into robots presents many challenges, including real-time data processing, ensuring accuracy in complex industrial environments, and calibrating cameras to synchronize with the robot coordinate system [7-8]. Issues such as heterogeneous software and hardware integration, high initial investment costs, and difficulties in handling complex objects (reflective, transparent) are also major barriers. In addition, optimizing algorithms to achieve high performance on limited hardware while maintaining system reliability and scalability also requires significant time and resources.

Robots often work in challenging industrial environments such as changing lighting, dust, obscured objects, or objects of various shapes and sizes. Maintaining the accuracy of vision systems under such conditions is a major problem. Traditional methods use image processing algorithms based on geometric and color features to identify objects. Therefore, the surrounding workspace has a great influence on the performance of the system. Traditional methods usually require a stable working environment with little change. Moreover, image processing methods are also limited to a few products, making it difficult for them to meet the requirements of modern industrial applications [9-11]. To overcome the limitations of traditional methods, deep learning has been applied to improve the recognition and processing capabilities of vision systems in complex industrial environments. With the ability to automatically extract features from image data without manual setup, deep learning models, especially convolutional neural networks (CNNs), can accurately recognize and classify objects, even under conditions such as changing lighting, occlusion, or complex shapes [12-14].

Deep Neural Networks for Object Detection provide the fastest and most accurate results for single and multiple object detection as CNNs can learn automatically with less manual effort [15-16]. Deep learning-based object detection models are classified into two classes: single-stage and two-stage detectors. Single-stage object detectors predict directly, eliminating the region proposal step. On the other hand, two-stage object detectors involve region proposals followed by classification and proposal refinement. The family of region-based CNN models is one of the most popular and advanced 2-phase architectures for object detection. Single-stage object detection methods aim to

Received: January 2025, Accepted: March 2025
Correspondence to: Le Hoai Phuong
Industrial Maintenance Training Center,
Ho Chi Minh City University of Technology (HCMUT)
E-mail: lhphuong@hcmut.edu.vn
doi: 10.5937/fme2502233P

simplify the object detection pipeline by predicting object class labels and bounding box coordinates in a single pass, thus often achieving faster processing speeds than two-stage methods. Given their efficiency, they are a popular choice for real-time object detection. Some prominent single-stage object detection models are YOLO, SSD, and RetinaNet. Among them, the YOLO model has become an outstanding model for real-time object detection [17-19]. The ability of YOLO to perform real-time object detection with reasonably good accuracy makes it versatile for a wide range of applications that require swift and accurate object recognition. In robotic applications, YOLO is used for object recognition and localization, enabling robots to perceive and interact with their environment more effectively. YOLO architecture is designed to optimize both detection speed and accuracy. This makes it particularly useful in robotics, where decisions often need to be made in milliseconds. Unlike traditional computer vision algorithms that may struggle with real-time processing, YOLO's deep learning-based approach enables the robot to process visual data rapidly, identify potential hazards, and react accordingly [20-21].

Most existing YOLO-based methods focus solely on 2D object detection and require additional post-processing techniques to estimate depth information. This limitation makes them less suitable for robotic manipulation tasks, where precise 3D position estimation is essential. To address this, various depth estimation techniques have been explored, including monocular depth estimation, LiDAR, Time-of-Flight (ToF) sensors, and stereo vision, each with its advantages and drawbacks.

Monocular depth estimation relies on a single 2D image to predict depth but often suffers from high estimation errors in real-world industrial settings, particularly when dealing with complex object geometries, non-uniform textures, or reflective surfaces. LiDAR and ToF cameras provide high-accuracy depth measurements but come with significant drawbacks, including high costs, limited resolution, and sensitivity to ambient lighting conditions, making them less viable for cost-sensitive industrial automation. In contrast, stereo vision-based depth estimation offers a cost-effective and real-time alternative by using two cameras to triangulate depth information. However, stereo vision systems can be prone to inaccuracies, particularly in cases where objects lack texture, are exposed to poor lighting, or experience significant occlusions. These challenges necessitate advanced depth filtering and calibration techniques to enhance accuracy in industrial applications.

To address these limitations, this study proposes an intelligent robotic pick-and-place system that integrates YOLOv11-based object detection with stereo vision-based 3D localization. The system utilizes an Intel RealSense D435 stereo camera to capture both RGB and depth data, enabling accurate 3D object localization and manipulation. Unlike previous approaches, our method seamlessly integrates deep learning-based object detection with stereo-depth estimation, ensuring robust and real-time performance in dynamic and unstructured industrial environments.

Stereo vision is a suitable choice in robotic pick-and-place applications. Within a range of several meters for manipulation, stereo vision systems provide an accurate measurement solution at a reasonable cost compared to solutions using other devices (LiDAR, ToF camera). Furthermore, stereo vision can operate in natural light conditions, thus helping robots work in normal lighting conditions of factory production lines [24-30]. In this paper, we use the Intel D435 camera for robotic pick-and-place applications. The system utilizes the camera to capture RGB images and depth images. An object detection model, based on YOLOv11, is employed to identify objects placed in a bin. The depth images are then processed to detect the objects and determine their 3D coordinates. Using this information, the robot moves its gripper to the object's position, picks it up from the bin, and places it on a conveyor. This approach is a common application in production lines, where objects are picked from containers and transported for processing, inspection, or packaging.

The proposed system has significant practical applications in industrial automation, particularly in automated sorting, packaging, and assembly processes. By leveraging a stereo vision-based 3D localization approach, the system achieves high precision in object detection and manipulation while maintaining cost efficiency compared to LiDAR or ToF-based solutions. This research contributes to the advancement of intelligent robotic vision systems, demonstrating an effective method for real-time vision-robot coordination in unstructured environments. The integration of deep learning-based object detection with depth estimation techniques enhances the accuracy and adaptability of robotic pick-and-place operations, making the system a valuable solution for modern smart manufacturing and Industry 4.0 applications.

2. MATERIAL AND METHODS

2.1 Robot arm system performance and characteristics

In this paper, the 4-DOF palletizing robot arm is developed to grasp objects in a bin and place them on the conveyor. Figure 1 describes the design of the robot arm. The robot arm is designed with four degrees of freedom and is driven by AC servo motors connected to harmonic gearboxes to create precise movements. The use of AC servo motors and harmonic gearboxes in the design improves the robot's performance and precision, and it can operate stably for a long time. AC servo motor provides precise control of position, speed, and torque thanks to the high-resolution encoder. With the zero backlash transmission mechanism, the harmonic gearbox minimizes vibration and position error, enhancing the precision of the robot. In addition, harmonic gearboxes provide a very large transmission ratio in a compact size, increasing output torque without using a larger motor. This helps reduce the overall size and weight of the robot. We use Yaskawa's 100w sigma 5 AC servo motor and a 1/50 harmonic gearbox. We chose aluminum alloy to manufacture the robot's joints to reduce weight, increase rigidity, and improve energy

efficiency. The lengths of the robot's arms are 450mm, 400mm, and 100mm respectively. This combination allows the robot to reach a maximum reach of 1050mm and carry a payload of up to 500g, meeting the high requirements for performance and stability in diverse applications. Table 1 shows the specifications of the robot arms.

Figure 1. Robot arm design

Table 1. Specifications of the robotic arm.

Specification	Value
Degree of freedom	4
Maximum reach	1050mm
Payload	4kg
Power supply	220V-500W
Weight	8kg
Actuator	100w AC servo motor
Gearbox	Harmonic drive
Max speed of joint 1	180^0/s
Max speed of joint 2	180^0/s
Max speed of joint 3	200^0/s
Max speed of joint 4	300^0/s

The robot's electrical and control system is designed with the PLC acting as the central controller, ensuring accuracy and stability in controlling the movement of the robot joints. The PLC generates high-speed pulse signals, transmitted to the AC servo drivers, thereby controlling the motor to rotate at the required angle. Thanks to the fast and reliable processing capabilities of the PLC, the system can accurately adjust the speed and position of the robot joints, ensuring fast and stable response to tasks. This combination not only helps optimize operating performance but also brings high reliability to the entire robot control system. The control system uses Delta's DVP28SV11T PLC (shown in Figure 2), a mid-range PLC with powerful performance and integrated features suitable for controlling 4-joint robots. This PLC is equipped with 4 high-speed output pins, ideal for accurately controlling the signal pulses sent to the AC servo driver, ensuring that the robot joints move at the desired angle and speed. In addition, the PLC has a built-in RS485 communication port, making it easy to communicate with computers and other industrial devices, supporting flexible connection and system control. Thanks to these features, the DVP28SV11T PLC not only meets the requirements for accuracy and speed but also provides superior connec–tivity, suitable for automatic control applications in modern industrial environments. Table 2 shows the technical specifications of Delta PLC.

Figure 2. PLC delta DVP28SV11T PLC

Table 2. Technical Specifications off PLC.

Specification	Value
Dimensions (L x W x H)	70 mm x 60mm x 90mm
Power Supply	24 VDC
Number of Input Pins	16
Number of Output Pins	12
Output Type	NPN
Working Capacity	16000 steps
Operating Temperature	0^0C to 55^0C
Memory Capacity	10000 words

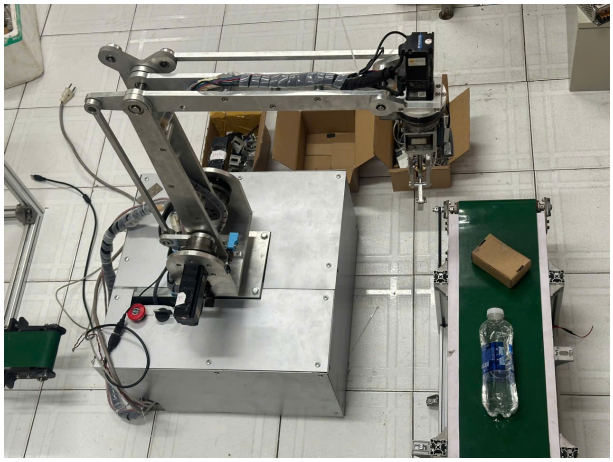

Figure 3. Practical system

Figure 3 shows the actual robot system that has been developed. The robot will pick up objects moving on the conveyor belt and put them into containers. The coordinates of the objects on the conveyor belt will be determined by the 3D vision system using the Intel D435 camera.

2.2 The 3D intelligence vision system

A 3D intelligent vision system is developed to provide the ability to accurately locate objects in three-dim–ensional space for robots to perform pick and place

operations. In this study, we use the Intel RealSense D435 camera, a high-performance stereoscopic vision device, to collect both RGB images and depth images of the working environment.

The Intel RealSense D435 camera operates on the stereoscopic principle, with the ability to provide accurate depth data within a few meters. The advantages of this camera are reasonable cost, stable operation in natural light conditions, and suitable resolution for industrial applications. The vision system uses RGB images to detect objects and depth information to determine the spatial coordinates of objects.

For object detection and classification, the YOLOv11 model is deployed thanks to its real-time processing capability and high accuracy. Once objects are identified in the RGB image, depth data is utilized to pinpoint the topmost object, reducing detection errors caused by occlusion or overlapping objects. The 3D coordinates of the detected object are then calculated and transmitted to the robot through RS485 communication, enabling precise positioning of the gripper for the gripping operation. The integration of the Intel RealSense D435 camera and the YOLOv11 model allows the vision system to deliver high performance in industrial applications. Notably, the system demonstrates stable operation in real production environments, even under changing or complex lighting conditions.

2.2.1 Data collection

The vision system uses an Intel RealSense D435 camera to collect data including RGB images and depth maps of the work area. The camera is fixed on a bracket above the container, with a panoramic view to observe the entire surface of the container.

The data collection process is carried out under different lighting conditions to simulate a variety of real-life environments and increase the generalization ability of the model. Furthermore, objects are placed in different positions and postures to ensure that the model learns to recognize in many situations.

In total, about 400 images were collected:

200 images: Contains only objects of interest, arranged neatly in the container to provide clear data for the learning model.

200 images: Includes additional objects placed around the target blocks, simulating a more messy and complex logistics environment.

This data not only increases diversity but also ensures that the system can operate effectively in real production lines, where the environment is often not completely clean. The captured images are used to train and evaluate the object detection model.

2.2.2 Data Annotation

After collecting the data, the data annotation process is performed to label the objects of interest in the image set. The Roboflow tool is used for this, thanks to its friendly interface and features that support accurate labeling. Objects are annotated by bounding boxes, and labeled according to specific categories to serve the object detection model training process.

To improve the performance and generalization ability of the model, data augmentation techniques are applied, including:

Rotation: Helps the model learn to recognize objects at different angles.

Brightness adjustment: Simulates diverse lighting conditions in reality.

Random cropping: Helps the model learn to handle incompletely displayed parts of the object.

Gaussian noise: Increases the model's tolerance to errors in noisy data conditions.

The labeled and augmented data is divided into two sets, 75% for the training set, used to train the object detection model, and 25% for the test set, used to evaluate the performance of the model after training.

2.2.3 Model training

To improve the object detection performance, the YOLOv11 model was trained using transfer learning on a custom dataset. The dataset consisted of 900 RGB images captured using the Intel RealSense D435 camera, representing three object classes: bottle, can, and cup. Each class contained approximately 300 images, ensuring a balanced distribution. The images were collected under varying lighting conditions and object arrangements to enhance the model's robustness. Annotation was performed using the Roboflow platform, where each object was labeled with bounding boxes and exported in a YOLO-compatible format. The dataset was then split into 75% for training and 25% for validation.

To improve generalization and prevent overfitting, various data augmentation techniques were applied. These included geometric transformations (random rotations, scaling, and flipping), photometric adjustments (brightness and contrast variations, Gaussian noise), and synthetic occlusion to simulate real-world cluttered environments. These augmentations aimed to improve the model's ability to recognize objects under different conditions and enhance its robustness in real-world applications.

The training process was conducted on Google Colab with GPU acceleration, leveraging the pre-trained YOLOv11 weights to reduce computational cost and training time. The model was fine-tuned using a batch size of 16, trained for 100 epochs, with an input image resolution of 640 × 480 pixels. The AdamW optimizer was employed, incorporating a cosine annealing learning rate scheduler to ensure stable convergence. The loss function used was CIoU loss for bounding box regression and focal loss for object classification, both of which helped improve detection accuracy for small and occluded objects.

Throughout the training process, real-time monitoring was performed to track key performance metrics, including loss, precision, recall, and mean Average Precision (mAP). The model was periodically evaluated on the validation set to assess its learning progress and detect any potential overfitting. This systematic approach to training ensured that the model adapted

effectively to the requirements of dynamic industrial environments, making it well-suited for real-time pick-and-place applications.

After the training process is completed, the best model (best.pt) is downloaded and deployed on the hardware to perform the task of detecting objects in camera images. The model is integrated into the robot's vision system, using RGB images from the Intel RealSense D435 camera as input. When detecting an object, the model outputs the coordinates of the rectangle surrounding the object, including information about the center coordinates, length, width, and confidence score. Only objects with a confidence score higher than 0.5 are accepted to ensure high accuracy during the detection process. This information is transmitted to the robot control system via RS485 communication, helping the robot accurately determine the location and size of the object to perform the pick-and-place operation. With fast processing speed and high accuracy, the YOLOv11 model has well met the system requirements in the real production environment. Figure 4 shows the flowchart to train the object detection model.

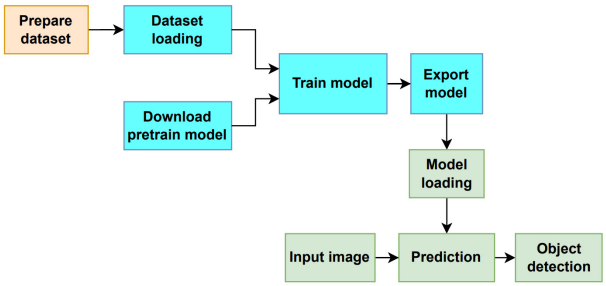

Figure 4. Flowchart to train the model for object detection

2.3 Robot control methodology

After determining the coordinates of the object in the RGB image through the YOLOv11 model, the vision system combines with the depth image data from the Intel RealSense D435 camera to calculate the three-dimensional (3D) coordinates of the object. The information including the coordinates of the center of the object on the image plane (X, Y) and the depth (Z) are used to determine the actual position of the object in three-dimensional space.

The 3D coordinates of the object are then converted to the coordinate system of the robot through the calibration process between the camera and the robot, ensuring synchronization between the axis systems. Next, the inverse kinematics problem is applied to convert the 3D coordinates of the object to the corresponding rotation angles of the robot joints. The inverse kinematics problem calculates the necessary motion parameters for each joint of the robot to bring the gripper to the correct position and orientation in space.

These rotation angle parameters are transmitted to the robot control system via the PLC controller, enabling the robot to perform smooth and precise movements. This process allows the robot to identify and pick up objects efficiently, meeting the requirements of accuracy and speed in industrial manufacturing applications. The combination of the vision system and the inverse kinematics problem not only improves performance but also expands the application capabilities of the robot in complex environments.

2.3.1 Coordinate System Conversion

Figure 5 illustrates the coordinate relations in the computer vision robotic system. The coordinate system $O_C X_C Y_C Z_C$ represents the camera coordinate system, while the coordinate system $O_R X_R Y_R Z_R$ refers to the robot's base coordinate system. To enable the robot to interact with objects based on visual data, it is necessary to convert the coordinates from the camera system to the robot system. This conversion can be mathematically described using the following equation:

$$\begin{bmatrix} X_R \\ Y_R \\ Z_R \end{bmatrix} = R \begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} + T \tag{1}$$

where R is the rotation matrix that represents the orientation of the camera relative to the robot's base, $\boldsymbol{T}$ is the translation vector that defines the displacement between the origin of the camera coordinate system and the robot coordinate system, $[X_R, Y_R, Z_R]^T$, and $[X_C, Y_C, Z_C]^T$ are the coordinates of the object in the robot and camera coordinate systems, respectively.

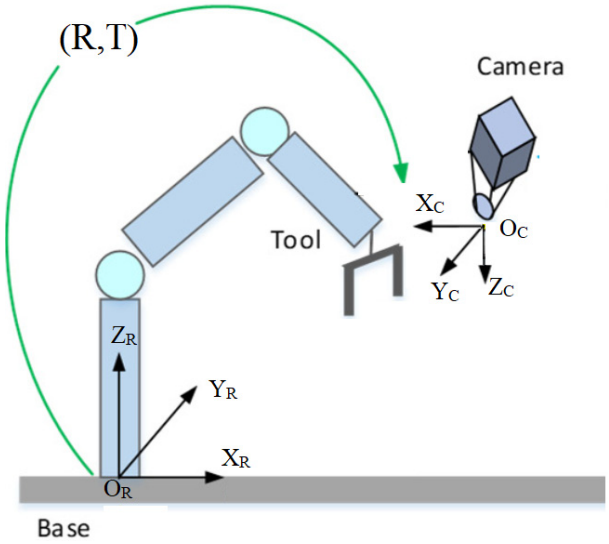

Figure 5. Robot and camera coordinate relation

To convert the pixel coordinates obtained from the camera image into the camera's 3D coordinate system, we use the intrinsic camera parameters. This relationship can be expressed using the following equation, which utilizes the camera's intrinsic matrix:

$$Z_C \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} \tag{2}$$

where $\boldsymbol{u}$ and v are the pixel coordinates of the object in the image, $\boldsymbol{f_x}$ and $\boldsymbol{f_y}$ are the focal lengths of the camera in the x and y directions (in pixels), $\boldsymbol{u_0}$ and $\boldsymbol{v_0}$ are the coordinates of the principal point (typically the image center), Z_C is the depth of the object in the camera's

coordinate system, which is obtained from the depth image or stereo vision system.

The equation above describes how to map pixel coordinates to 3D coordinates in the camera frame. To convert from pixel coordinates (U,V) to camera coordinates $[X_C, Y_C, Z_C]^T$, the inverse of the camera's intrinsic matrix can be applied, with depth Z_C as a scaling factor:

$$X_C = \frac{u - u_0}{f_x} Z_C$$
$$Y_C = \frac{v - v_0}{f_y} Z_C \tag{3}$$

2.3.2 Motion control using PLC commands

In the motion control system of the robot, the WPLSoft software is used to program and configure Delta PLCs for controlling the robot's movements. This software, developed by Delta Electronics, is lightweight and features a user-friendly interface compared to other PLC programming tools such as Mitsubishi's GX Works or ISPSoft. It supports programming for a wide range of Delta PLC models and allows for easy integration with robotic systems.

One of the key commands for controlling motor movement in Delta PLC systems is the DDRVI com–mand, which is used to generate output pulses that control the speed and direction of motors. This com–mand is specifically designed for controlling stepper motors, servo motors, or other devices that operate based on pulse signals.

The DDRVI command structure is as follows:

$$DDRVI\ S1 S2\ D1 D2 \tag{4}$$

where:

S1: Represents the position pulse for the SV2 series motor. This 32-bit value specifies a range from -2,147,483,648 to +2,147,483,647. If the value of S1 is 0, it indicates that no output will be generated, and no action will take place.

S2: Specifies the speed pulse for the SV2 series motor. This 32-bit value can be set within the range of 0 to 200,000 Hz.

D1: Specifies the pulse output pin. For the SV2 series, the pulse output can be directed to pins Y0, Y2, Y4, or Y6.

D2: Determines the direction of the pulse. The action of D2 depends on the sign of S1. If S1 is negative, D2 will be turned off. It will not immediately turn off after the pulse output ends; it will only turn off when the control contact specified by the command is deactivated.

To convert the joint angles into the necessary number of pulses, the inverse kinematics calculations first provide the required angles for each joint. These angles are then converted into pulses by considering the step angle of the motor:

$$Pulse = \frac{Joint\ Angle}{ANGLER\ PER\ PULSE} \tag{5}$$

Each motor's rotation is divided into a number of discrete steps (pulses). The angle corresponding to one pulse, also known as the "pulse resolution," depends on the motor's specifications and the gear ratio of the joint.

$$Angle\ per\ pulse = \frac{360°}{N.o.\ pulses\ per\ revolution} \tag{6}$$

The DDRVI command also supports smooth acceleration and deceleration of the motor's movement by adjusting the time parameters for these phases. The time required for acceleration and deceleration can be configured for each output pin (Y0, Y2, Y4, Y6) using specific data registers, ensuring smooth transitions during movement. The registers for adjusting acceleration and deceleration times are as follows:

D1343: Configures the acceleration and deceleration time for the pulse output on Y0.

D1353: Configures the acceleration and deceleration time for the pulse output on Y2.

D1381: Configures the acceleration and deceleration time for the pulse output on Y4.

D1382: Configures the acceleration and deceleration time for the pulse output on Y6.

By adjusting these parameters, the robot's motors accelerate smoothly to the desired speed and decelerate gradually to avoid jerky movements, which can cause mechanical stress or inaccuracies. This capability en–hances both the performance and longevity of the system.

3. EXPERIMENT RESULTS

Figure 6 illustrates the training results for the YOLOv11 model, demonstrating the loss reduction and perfor–mance metrics over 100 epochs. The graphs indicate significant convergence in training and validation losses, alongside improvements in precision, recall, and mAP metrics. Table 3 summarizes the quantitative results of the model's performance on the test dataset, providing a breakdown of precision, recall, mAP@50, and mAP@50-95 for all classes and specific object categories (bottle, can, cup). The results demonstrate that the model achieves high detection performance across all categories, with particularly strong recall values for the "bottle" class and near-perfect precision for the "cup" class. The mAP@50 score of 0.985 signifies robust detection capabilities, while the mAP@50-95 score of 0.942 confirms the model's ability to generalize across varying intersection-over-union (IoU) thresholds. This performance establishes the effectiveness of the YOLOv11 model in object detection tasks for industrial applications.

Table 3. Performance metrics for object detection model.

Class	precision	recall	mAP50	mAP50-95
All	0.973	0.987	0.985	0.942
bottle	0.955	1.000	0.971	0.939
can	0.970	0.975	0.994	0.942
cup	0.993	0.986	0.990	0.945

To evaluate the performance of the YOLOv11 model in real-world scenarios, object detection was conducted under various conditions, including different backgrounds, object positions, and angles.

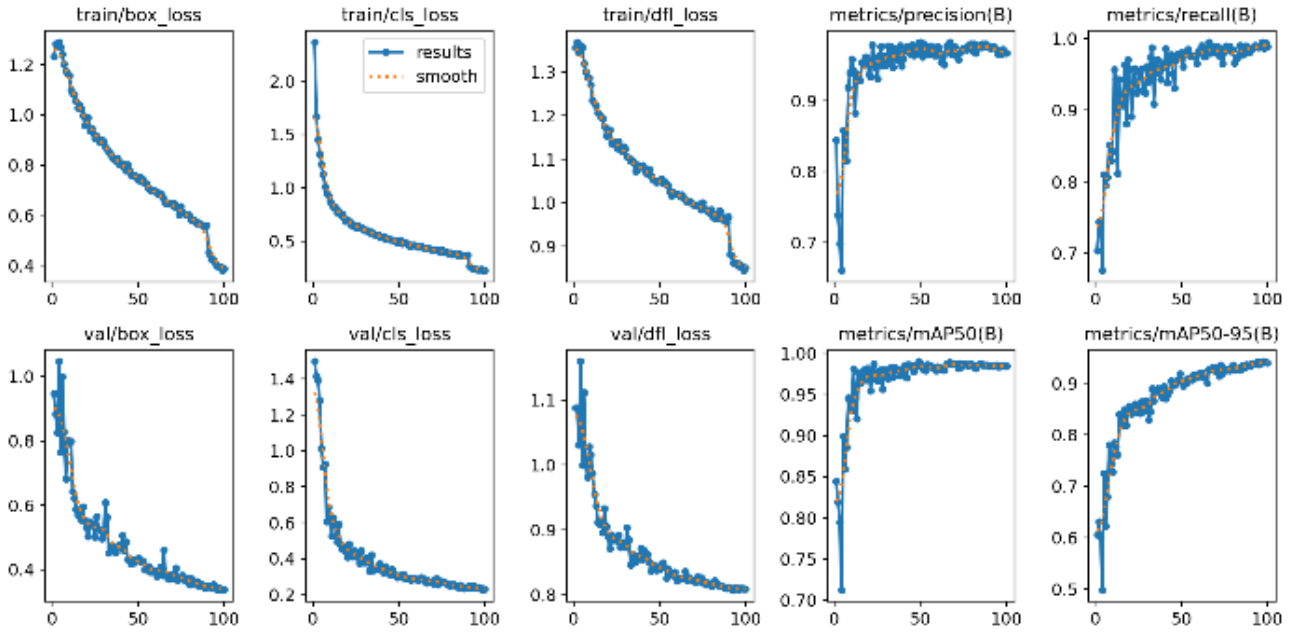

Figure 6. The training results for the YOLOv11 mode

The evaluation included images containing one or multiple target objects alongside other unrelated items. As illustrated in Figure 7, the model consistently demonstrated high accuracy in detecting and classifying objects with confidence scores exceeding 0.8. Despite variations in environmental factors, such as object orientation or the presence of occlusions, the YOLOv11 model effectively localized and identified objects such as bottles, cans, and cups. These results highlight the robustness and adaptability of the model, making it suitable for applications in complex and dynamic environments. The capability to maintain high precision and recall across diverse conditions further emphasizes the potential for integrating the system into industrial automation workflows.

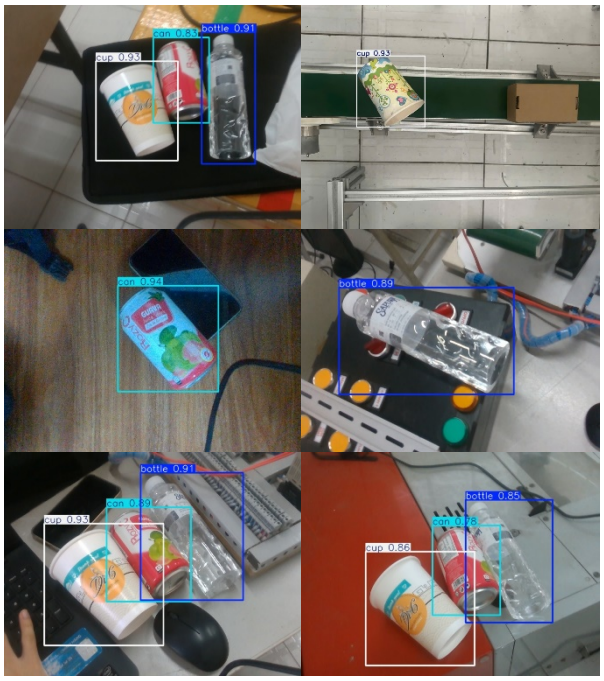

Figure 7. The detection and classification results

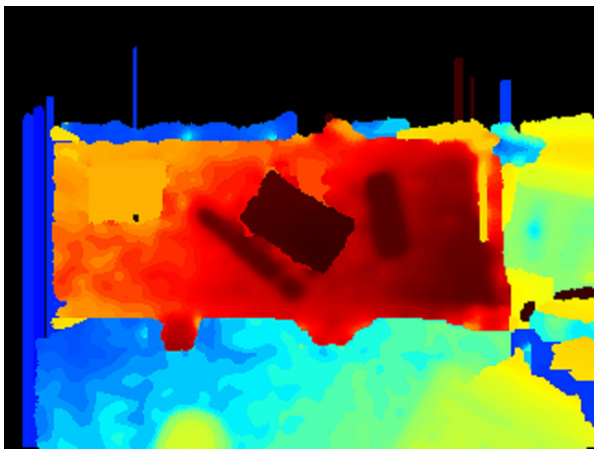

Figure 8. The depth image was taken by the D435 camera

Once the objects have been located in the RGB image, the depth values extracted from the depth image are used to determine the 3D coordinates of the objects. Figure 8 shows the depth image taken from the D435 camera. After being aligned, the pixel coordinates of the objects in the RGB image and the depth image will be the same. Therefore, from the centroid coordinates of the object in the RGB image, we extract an ROI region to get the depth value of the object.

Table 4. Performance metrics for 3D coordinate calculation.

Object	Errors (mm)		
	X	Y	Z
bottle	2.18	2.35	4.5
can	2.20	2.41	4.8
cup	2.51	2.44	5.3

Table 4 shows the accuracy performance of the 3D vision system to calculate the position of three objects. It can be seen that the largest error is always at the Z coordinate (depth value), ranging from 4.5 mm to 5.3 mm depending on the object. The error in Z coordinates (depth value) when using depth cameras is often quite large, especially when measuring objects with curved surfaces.

The main reason is that the sensor receives many different depth values at points on the surface, and when taking the average value to extract the depth, this method cannot accurately reflect the true shape of the curved surface. In addition, the limited resolution of the sensor and noise from the environment or the device itself also contribute to the error. Although averaging helps reduce noise, it loses detailed information in curved or non-uniform areas. To minimize the error, advanced processing algorithms can be applied such as smoothing or 3D surface reconstruction, increasing the resolution of the camera, or using advanced filters such as Gaussian or median filters to reduce noise while preserving information. The error of the Z coordinate (depth value) not only directly affects the accuracy of the depth value but also leads to errors in the calculation of the X and Y coordinates. Since the X and Y coordinates are usually derived from the Z value through projections and formulas related to the camera's field of view, any deviation in the Z value will propagate and cause errors in determining the position on the X-Y plane. For the X and Y coordinates, the error ranges from 2.18 mm to 2.51 mm, which is much smaller than the Z error.

4. CONCLUSIONS

In this study, we developed a 4-degree-of-freedom pick-and-place robotic system integrated with a high-precision control mechanism and an intelligent 3D vision system. The robotic system employs an AC servo motor combined with a harmonic gearbox, ensuring high accuracy and stability in industrial environments. The vision system, built upon an Intel RealSense D435 depth camera and a YOLOv11-based object detection model, enables real-time 3D localization of objects with high precision. The experimental results demonstrate that the proposed system achieves robust performance, with the YOLOv11 model obtaining a mAP@50 of 0.985 and a mAP@50-95 of 0.942, ensuring reliable object detection in complex, dynamic environments.

A key novelty of this research lies in the integration of stereo vision-based depth estimation with real-time object detection and robotic manipulation, which enhances the system's ability to operate in unstructured industrial settings. The proposed coordinate system processing and correction techniques improve synchronization between the vision system and the robotic arm, ensuring precise and stable picking operations. Experimental evaluation of the 3D coordinate calculation shows that the system achieves an average localization error of 2.29 mm in the X-axis, 2.40 mm in the Y-axis, and 4.87 mm in the Z-axis. These errors are within an acceptable range for industrial pick-and-place applications, demonstrating the system's suitability for real-world deployment.

Future research will focus on further improving depth estimation accuracy, integrating higher-resolution sensors, and optimizing error compensation algorithms to enhance the system's adaptability to more complex industrial environments.

ACKNOWLEDGMENT

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

REFERENCES

[1] Shahin, M., Chen, F.F., Hosseinzadeh, A. et al. Ro–botics multi-modal recognition system via compu–ter-based vision. Int J Adv Manuf Technol (2024). https://doi.org/10.1007/s00170-024-13164-z.

[2] L.H. Phuong, V.D. Cong, "Control the robot arm through vision-based human hand tracking", FME Transactions, Vol. 52, No. 1, pp. 37 – 44, 2024.

[3] Nguyen, T., & Vo Duy, C. (2024). Grasping moving objects with incomplete information in a low-cost robot production line using contour matching based on the Hu moments. *Results in Engineering*, *23*, 102414. https://doi.org/10.1016/j.rineng.2024.102414

[4] LP. Nguyen, H.Q.T Ngon, Framework Design using the Robotic Augmented Reality for the CyberPhysical System, FME Transactions (2024) 52, 506-516.

[5] Nussibaliyeva, A., Sergazin, G., Tursunbayeva, G., Uzbekbayev, A., Zhetenbayev, N., Nurgizat, Y., Bakhtiyar, B., Orazaliyeva, S., & Yussupova, S. (2023). Development of an Artificial Vision for a Parallel Manipulator Using Machine-to-Machine Technologies. Sensors, 24(12), 3792. https://doi.org/10.3390/s24123792

[6] Chen, Y., Cai, Y., Cheng, M. (2023). Vision-Based Robotic Object Grasping—A Deep Reinforcement Learning Approach. *Machines*, *11*(2), 275. https://doi.org/10.3390/machines11020275

[7] Jahanshahi, H., Zhu, Z. H. (2024). Review of machine learning in robotic grasping control in space application. Acta Astronautica, 220, 37-61. https://doi.org/10.1016/j.actaastro.2024.04.012

[8] C. Yuanwei, M. Hairi Mohd Zaman and M. Faisal Ibrahim, "A Review on Six Degrees of Freedom (6D) Pose Estimation for Robotic Applications," in IEEE Access, vol. 12, pp. 161002-161017, 2024

[9] Li, C., Dun, X., Li, L. et al. Vision-guided robot application for metal surface edge grinding. SN Appl. Sci. 5, 236 (2023). https://doi.org/10.1007/s42452-023-05468-8

[10] Elassal, A., Abdelaal, M., Osama, M., & Elhnidy, H. (2024). Low-cost parallel delta robot for a pick-and-place application with the support of the vision system. E-Prime - Advances in Electrical Engi–neering, Electronics and Energy, 8, 100518. https://doi.org/10.1016/j.prime.2024.100518

[11] H.V. Nguyen, V.D. Cong, P.X. Trung, Develop–ment of a SCARA robot arm for palletizing applications based on computer vision, FME Transaction, Vol. 51, No. 4, pp. 541-549, 2023.

[12] Chen, Y., Cai, Y., Cheng, M. (2023). Vision-Based Robotic Object Grasping—A Deep Reinforcement Learning Approach. Machines, 11(2), 275. https://doi.org/10.3390/machines11020275

[13] D'Avella, S., Tripicchio, P., & Avizzano, C. A. (2020). A study on picking objects in cluttered en–vironments: Exploiting depth features for a custom low-cost universal jamming gripper. Robotics and

Computer-Integrated Manufacturing, 63, 101888. https://doi.org/10.1016/j.rcim.2019.101888

[14] D'Avella, S., Avizzano, C. A., & Tripicchio, P. (2023). 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, 80, 102453. https://doi.org/10.1016/j.rcim.2022.102453

[15] Pietrala, D. S., Laski, P. A., Zwierzchowski, J., Borkowski, K., Bracha, G., Borycki, K., Kostecki, S., Wlodarczyk, D. (2022). Autonomous Manipu–lator of a Mobile Robot Based on a Vision System. *Applied Sciences*, *13*(1), 439. https://doi.org/10 .3390 /app13010439

[16] Ribeiro, E. G., De Queiroz Mendes, R., Grassi, V. (2021). Real-time deep learning approach to visual servo control and grasp detection for autonomous robotic manipulation. *Robotics and Autonomous Systems*, *139*, 103757. https://doi.org/10.1016/ j.robot.2021.103757

[17] Song, Q., Li, S., Bai, Q., Yang, J., Zhang, X., Li, Z., Duan, Z. (2021). Object Detection Method for Grasping Robot Based on Improved YOLOv5. Micromachines, 12(11), 1273. https://doi.org/10. 3390/mi12111273

[18] A. Maideena, A. Mohanarathinam, Computer Vision-Assisted Object Detection and Handling Framework for Robotic Arm Design Using YOLOV5, Advances in Distributed Computing and Artificial Intelligence Journal, Vol. 12 No. 1 (2023)

[19] Tella, H., Mohandes, M. A., Liu, B., Al-Shaikhi, A., Rehman, S. (2024). A Novel Cost-Function for Tra–nsformer-based YOLO Algorithm to Detect Pho–tovoltaic Panel Defects. FME Transactions, 52(4), 639-646. https://doi.org/10.5937/fme2404 639T.

[20] Wang, Y., Zhou, Y., Wei, L., & Li, R. (2022). Design of a Four-Axis Robot Arm System Based on Machine Vision. *Applied Sciences*, *13*(15), 8836. https://doi.org/10.3390/app13158836

[21] Varna, D., & Abromavičius, V. (2021). A System for a Real-Time Electronic Component Detection and Classification on a Conveyor Belt. Applied Sciences, 12(11), 5608. https://doi.org/10.3390/ app12115608

[22] Mu, X., Kan, Q., Jiang, Y., Chang, C., Tian, X., Zhou, L., Zhao, Y. (2025). 3D Vision robot online packing platform for deep reinforcement learning. Robotics and Computer-Integrated Manufacturing, 94, 102941. https://doi.org/10.1016/j.rcim.2024. 102941

[23] J. Jia, H. Shang and X. Chen, "Robot Online 3D Bin Packing Strategy Based on Deep Reinforcement Learning and 3D Vision," 2022 IEEE International Conference on Networking, Sensing and Control (ICNSC), Shanghai, China, 2022, pp. 1-6, doi: 10.1109/ICNSC55942.2022. 10004170.

[24] Shokhikha Amalana Murdivien, Jumyung Um, "BoxStacker: Deep Reinforcement Learning for 3D Bin Packing Problem in Virtual Environment of Logistics Systems", Sensors, vol.23, no.15, pp. 6928, 2023.

[25] Luis Ribeiro, Anan Ashrabi Ananno, "A Software Toolbox for Realistic Dataset Generation for Tes–ting Online and Offline 3D Bin Packing Algo–rithms", Processes, vol.11, no.7, pp.1909, 2023.

[26] Xiong, H., Ding, K., Ding, W., Peng, J., Xu, J. (2023). Towards reliable robot packing system ba–sed on deep reinforcement learning. *Advanced Engineering Informatics*, *57*, 102028. https://doi. org/10.1016/j.aei.2023.102028

[27] Benedek, C., Majdik, A., Nagy, B., Rozsa, Z., Szi–ranyi, T. (2021). Positioning and perception in LIDAR point clouds. *Digital Signal Processing*, *119*, 103193. https://doi.org/10.1016/j.dsp.2021. 103193

[28] R. Monica and J. Aleotti, "Point Cloud Projective Analysis for Part-Based Grasp Planning," in IEEE Robotics and Automation Letters, vol. 5, no. 3, pp. 4695-4702, July 2020.

[29] Shamsul Fakhar Abd Gani, Muhammad Fahmi Miskon, Rostam Affendi Hamzah, "Depth Map Information from Stereo Image Pairs using Deep Learning and Bilateral Filter for Machine Vision Application", 2022 IEEE 5th International Symposium in Robotics and Manufacturing Automation (ROMA), pp.1-6, 2022.

[30] Wang, H., Chen, C., Liu, Y., Ren, B., Zhang, Y., Zhao, X., & Chi, Y. (2024). A novel approach for robotic welding trajectory recognition based on pseudo-binocular stereo vision. *Optics & Laser Technology*, *174*, 110669. https://doi.org/10.1016/ j.optlastec.2024.110669

[31] Shu, Y., Zheng, W., Xiong, C., & Xie, Z. (2024). Research on the vision system of lychee picking robot based on stereo vision. *Journal of Radiation Research and Applied Sciences*, *17*(1), 100777. https://doi.org/10.1016/j.jrras.2023.100777

[32] Sumetheeprasit, B., Rosales Martinez, R., Paul, H., Shimonomura, K. (2023). Long-Range 3D Recon–struction Based on Flexible Configuration Stereo Vision Using Multiple Aerial Robots. *Remote Sensing*, *16*(2), 234. https://doi.org/10. 3390/rs 16020234

РАЗВОЈ ИНТЕЛИГЕНТНЕ ВИЗИЈЕ ЗА РОБОТСКИ СИСТЕМ ЗА БИРАЊЕ И ПОСТАВЉАЊЕ ОБЈЕКАТА

Л.Х. Фуонг, Ф.К. Трунг, Ф.Т. Кујен, Т.Т.Т. Маи

Овај рад представља аутоматизовани роботски систем pick-and-place који користи технологију стерео визије за детекцију и локализацију објеката у 3Д простору. Стерео визија је оптималан избор за индустријске апликације кратког домета због своје способности да обезбеди тачна мерења дубине по разумној цени, надмашујући алтернативе као што су ЛиДАР или Time-of-Flight (ToФ) камере у сличним

подешавањима. Предложени систем је дизајниран да поуздано ради у условима природног осветљења, што га чини веома погодним за примену у фабричким производним линијама. Интел РеалСенсе Д435 камера се користи за снимање РГБ и дубинских слика из окружења. Детекција објеката се врши коришћењем модела заснованог на ИОЛОв11, чиме се постиже висока тачност детекције са средњом просечном прецизношћу (мАП50) од 98,5% у свим класама објеката. Систем обрађује информације о дубини да би идентификовао највиши објекат, процењује његове 3Д координате са минималним грешкама (просечне позиционе грешке испод 5,3 мм) и преноси податке роботском манипулатору за извршење задатка бирања и постављања. Експериментални резултати показују високу прецизност и поузданост система у детекцији објеката и 3Д локализацији.