Mechanical Engineering Advances 2025, 3(2), 2474.
https://doi.org/10.59400/mea2474

Article

Design method for intelligent robots applied to traditional CNC processing plants: An integrated system based on mechanical, circuit, and image recognition technologies

Yung-Hsiang Chen[*], **Sheng-Yan Pan**

Department of Mechanical Engineering, National Pingtung University of Science and Technology, Pingtung 912301, Taiwan
*** Corresponding author:** Yung-Hsiang Chen, yhchen@mail.npust.edu.tw

CITATION

Chen YH, Pan SY. Design method for intelligent robots applied to traditional CNC processing plants: An integrated system based on mechanical, circuit, and image recognition technologies. Mechanical Engineering Advances. 2025; 3(2): 2474.
https://doi.org/10.59400/mea2474

ARTICLE INFO

Received: 30 December 2024
Accepted: 28 February 2025
Available online: 7 April 2025

Abstract: This study aims to design an automated production assistance device for small to medium-sized traditional CNC factories. The goal is to provide a cost-effective auxiliary production tool that integrates seamlessly into existing machining environments. The design encompasses mechanical, circuit, and software components. Mechanically, the device features a robotic arm equipped with a camera for object recognition and gripping, utilizing real-time image processing to enhance efficiency and stability. The circuit design employs embedded devices and microcontrollers to create a low-power, high-performance control system that manages motor drive, sensor data collection, and image recognition. On the software front, the system uses OpenCV and You Only Look Once (YOLO) for object detection and identification to tackle complex industrial scenarios. The design also considers economic feasibility, making it suitable for effective application in small and medium-sized enterprises. Through detailed theoretical analysis and multi-stage system simulations, the intelligent robot system has been thoroughly validated for overall stability and practicality. The final product is an intelligent self-propelled cart with capabilities, supporting efficient automated production and the intelligent upgrade of traditional manufacturing industries. Such a system is expected to significantly enhance production line efficiency in variable environments, reduce reliance on manual labor, and promote the intelligent transformation of traditional factories.

Keywords: mechanical design; circuit design; software design; YOLO; image recognition

1. Introduction

In the context of rapid industrial automation, continuous technological innovation has driven the deep integration of robotics, autonomous vehicles, and artificial intelligence (AI). This integration is pivotal in enhancing manufacturing efficiency, flexibility, and productivity. Traditional factories heavily rely on manual operations and fixed processes, leading to inefficiencies, lack of flexibility, and low precision, especially in complex production environments prone to human error [1–4].

According to market research data, the global market size of the CNC industry is expected to be approximately USD 100.7 billion in 2024 and is projected to grow at a compound annual growth rate (CAGR) of 7.8% until 2029. This growth trend indicates that with the continuous upgrading of global manufacturing and the advancement of Industry 4.0, the demand for industrial automation will keep expanding. To meet the increasing market demand, the application of CNC machines will continue to rise, further driving the rapid development of smart manufacturing and automation technologies.

Despite the potential for CNC machining plants to adopt automation equipment, the high costs and maintenance expenses remain prohibitive. Consequently, small to medium-sized enterprises (SMEs) struggle to afford these investments, posing significant challenges to their technological upgrades and automation processes. Additionally, the complexity of existing equipment and the lack of tailored solutions for SMEs hinder their ability to improve efficiency and reduce labor dependency. Therefore, this study proposes a design methodology based on mechanical, circuit, and image recognition techniques to create cost-effective and easy-to-maintain equipment.

Intelligent robots integrate multiple cutting-edge technologies, including robotic arms, autonomous vehicles, and image recognition systems, to complete complex tasks without human intervention. These systems utilize technological synergy to adapt to changing production requirements and execute tasks precisely. Moreover, these robots can collaborate with other automated equipment, moving the manufacturing environment closer to intelligent goals [5,6]. For example, robotic arms are designed to balance performance and safety, where human-friendly drive technology and high-precision calibration methods significantly enhance operational stability and accuracy [7,8]. For the needs of object detection and localization, deep learning provides advantages in real-time processing and small object recognition, offering strong support for intelligent robot applications [9–11].

This study proposes an intelligent robot system aimed at providing efficient, flexible, and sustainable solutions for small to medium-sized factories. The core design of this system is based on a six-degree-of-freedom robotic arm, paired with an autonomous vehicle for material transport, integrated with advanced image recognition technology for positioning and operations [12–14]. Such a design demonstrates superior adaptability in dynamic production environments, ensuring high operational accuracy and stability, thus achieving precise production control [15–17].

This research will delve into the mechanical design, circuit design, and software development of the intelligent robot, showcasing its application value and future development prospects in industrial environments. Through this study, we aim to provide practical references and technical support for the advancement of intelligent industrial development [18–20].

2. Intelligent robot design

The design of the intelligent robot integrates robotic arms, self-propelled vehicles, and image recognition technology to enhance the flexibility, efficiency, and intelligence of factory operations. The system can autonomously operate in complex environments and collaborate with other factory equipment to achieve a high level of integration between smart manufacturing and production lines.

2.1. Mechanism design

The system architecture of the intelligent robot is divided into three key components: a robotic arm, a mobile platform (self-propelled vehicle), and a visual recognition system. Each component plays a crucial role in the overall system, collaborating to achieve efficient and precise tasks.

2.1.1. The robotic arm design of the intelligent robot

The design of the robotic arm features a lightweight structure, with the main frame made of aluminum alloy 6061-T6, which offers high load-bearing capacity and rigidity. As shown in **Figure 1**, the structural design not only enhances stability but also ensures flexible operation under high loads. To improve control accuracy and power output, the mechanism incorporates a harmonic reducer (**Figure 2**) and a time-gauge pulley transmission system. The harmonic reducer, known for its high precision, small size, and high transmission efficiency, combines with the stable power transmission and seismic performance of the time-gauge pulley to ensure more accurate and stable overall mechanism operation. The design is based on a six-degree-of-freedom mechanism, enabling the robotic arm to have a wide operating range and high flexibility, capable of performing various precision operations according to actual needs. The arm's movement is driven by a precise servo motor, ensuring efficient and precise operation.

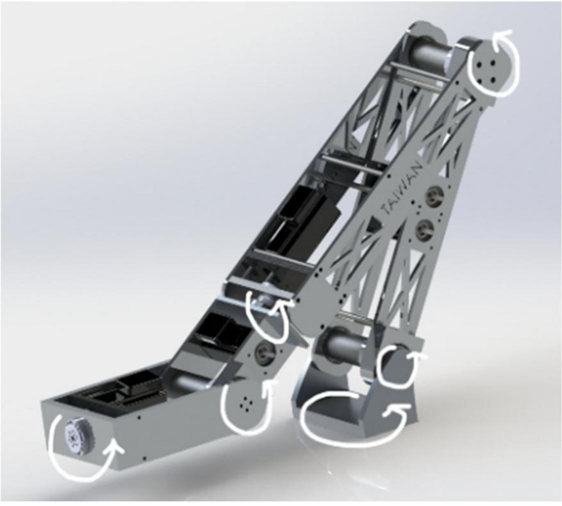

Figure 1. Six-axis robotic arm design schematic diagram.

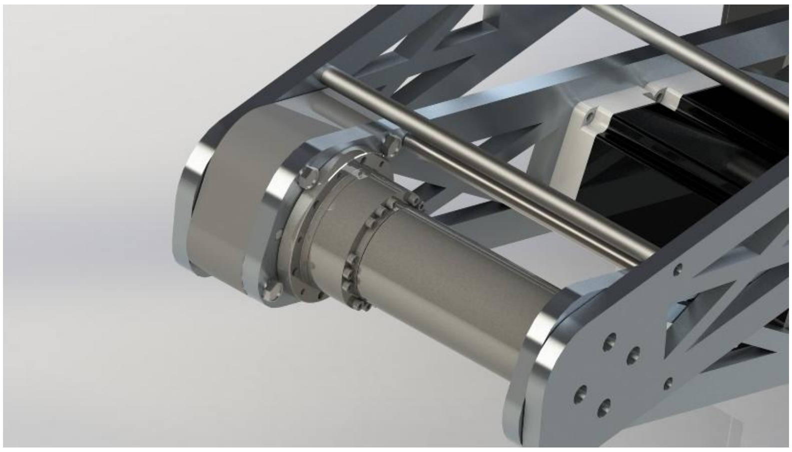

Figure 2. Schematic diagram of harmonic reducer.

The use of a harmonic reducer and timing belt transmission can obtain the following advantages at the same time:

- Effectively reduces shock and absorbs vibration, improving reliability.

- Power can be transmitted over a long distance, and at the same time, it can be decelerated by relying on harmonic reducers.
- High degree of freedom in structural design.
- High-precision transmission capability.

2.1.2. Design of self-propelled vehicles for intelligent robots

The design of the self-propelled vehicle mechanism in this study employs welding techniques to fabricate the structure, as shown in **Figure 3**. This approach aims to create a high-strength and integrated chassis. The application of welding technology ensures the chassis has superior structural integrity and durability. Even when the self-propelled vehicle carries various loads during operation, its structure remains stable, avoiding deformation or damage due to external forces or load changes. This chassis design not only enhances the rigidity and reliability of the overall mechanism but also effectively extends the service life of the self-propelled vehicle to meet the needs of diverse and complex environments.

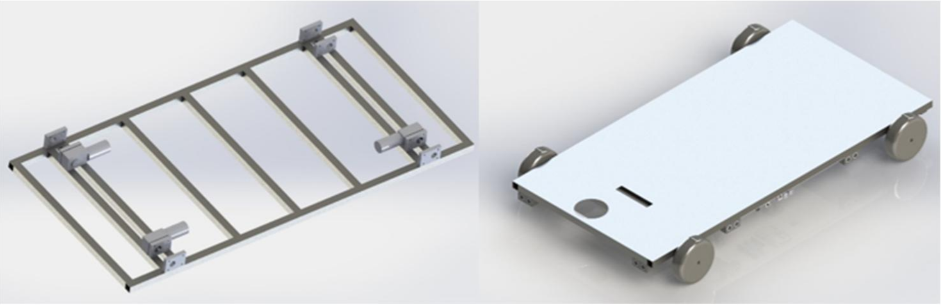

Figure 3. Self-propelled vehicle chassis structure diagram.

2.2. Circuit design of intelligent robot

In this study, multiple electronic components were utilized to form the overall system architecture, as shown in **Figure 4**. Given that image recognition applications are employed in both the robotic arm and the self-propelled vehicle mechanisms, two sets of Raspberry Pi are used to control the movement of the self-propelled vehicle and the robotic arm, respectively. These Raspberry Pi units can communicate with each other to achieve collaborative operation.

In the self-propelled vehicle mechanism, an Arduino acts as an intermediary, receiving data from ultrasonic sensors, radars, electronic compasses, and GPS. This data is packaged and transmitted to the Raspberry Pi for analysis, which then issues action commands back to the Arduino to drive motor movements. Once the movement command is executed and the destination is reached, communication is established between the Raspberry Pi of the self-propelled vehicle and the Raspberry Pi of the robotic arm. The next task is subsequently completed by the Raspberry Pi controlling the robotic arm, which is also equipped with a camera to detect the position of the workpiece, enabling precise movements.

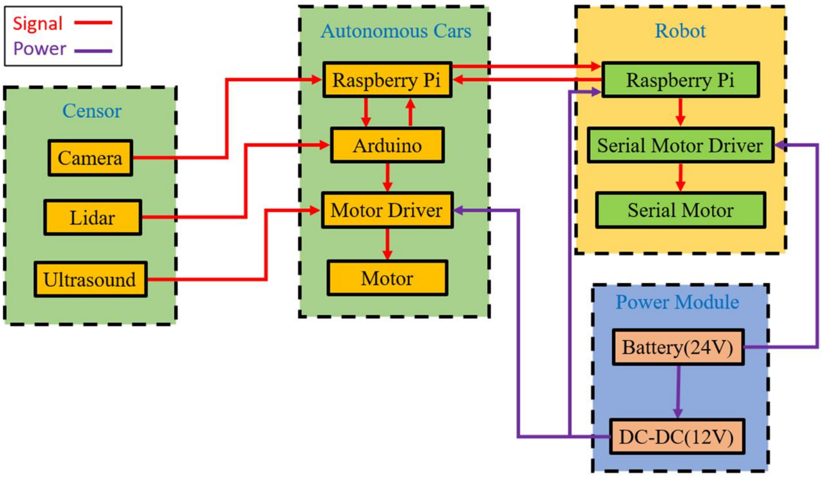

Figure 4. Overall system architecture of intelligent robots.

2.3. Software design of intelligent robot

The control system in this study primarily utilizes Raspberry Pi running Python as the core controller, responsible for coordinating the operations of the robotic arm and the self-propelled vehicle. The Raspberry Pi collects environmental data through sensors, conducts real-time path planning, and controls the motor to perform actions based on the path instructions. Additionally, it manages the servo motor to lift the robotic arm for picking up or placing down workpieces.

2.3.1. Control design

The entire system is illustrated in **Figure 5**, with a Raspberry Pi serving as the core controller, responsible for coordinating the operations of the robotic arm and the self-propelled vehicle. The Raspberry Pi collects environmental data through sensors, performs real-time calculations and path planning, and sends instructions to the servo motors and drive modules to execute operations. The workflow of the system is as follows: First, the self-propelled vehicle receives the destination information and begins navigation. After reaching the designated position, the robotic arm commences materials. Upon completing the task, both the self-propelled vehicle and the robotic arm return to their initial positions, ready to perform the next task.

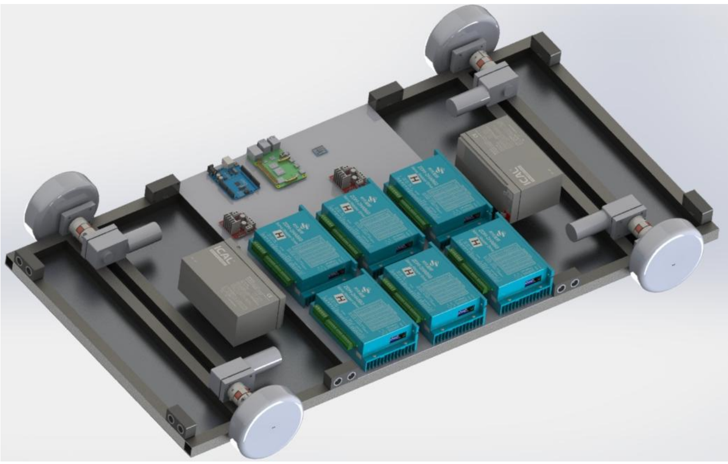

Figure 5. Intelligent robot control parts architecture diagram.

2.3.2. System operation process design

The system operation process of this study is divided into two parts: the system operation of the self-propelled vehicle and the robotic arm, and the interoperability is achieved through serial communication, as shown in **Figure 6**, and the overall process is as follows:

a) The operation process of the self-propelled vehicle:
  - Receive Navigation Destination: The self-propelled vehicle receives the navigation destination set by the user.
  - Path planning: Sensors (such as ultrasonic sensors, radars, electronic compasses, and GPS) are used to sense the environment, and the Raspberry Pi executes Python to set the walking path.
  - Perform the task: follow the planned path and receive sensor data to change direction and adjust speed
  - Arrive at the destination: Complete the task and notify the robot arm to perform the task.

b) The operation process of the robotic arm:
  - Task Receiving: Receive the task execution command sent by the Shumei faction and start executing the task.
  - Image inspection: The position of the workpiece is detected by the camera to ensure that it can be accurately positioned.
  - Grip and Place: Control the servo motor to operate the robotic arm to grip or lower the workpiece, and continuously use the camera to compensate for the position to achieve more accurate positioning in the process.
  - Complete the task: When the task is over, a command is sent to notify the self-propelled car Shumei faction to execute the next command.

c) Collaborative work:
  - The self-propelled vehicle and the robotic arm are synchronized through serial communication to ensure the continuity of navigation and tasks.
  - After executing a work, return to the initial position and wait for the next task order, which is distributed by the two Shumei factions.

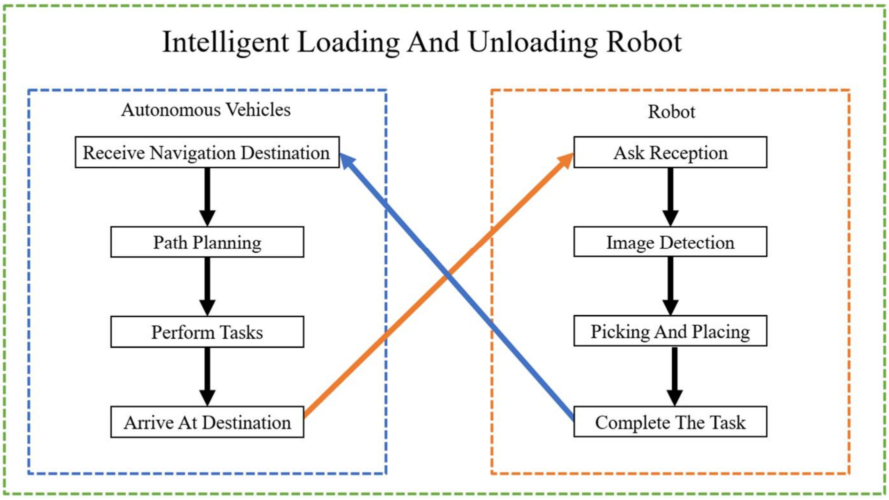

Figure 6. Flow chart of the operation of the intelligent robot.

2.3.3. Image recognition process

The image recognition system is a core function of the intelligent robot, primarily used to detect and identify the position and characteristics of workpieces. This ensures that the robotic arm can accurately perform operations. The system employs the YOLO (You Only Look Once) algorithm for real-time object detection and integrates OpenCV for image preprocessing and feature analysis. The entire image recognition process can be divided into the following stages, as illustrated in **Figure 7**:

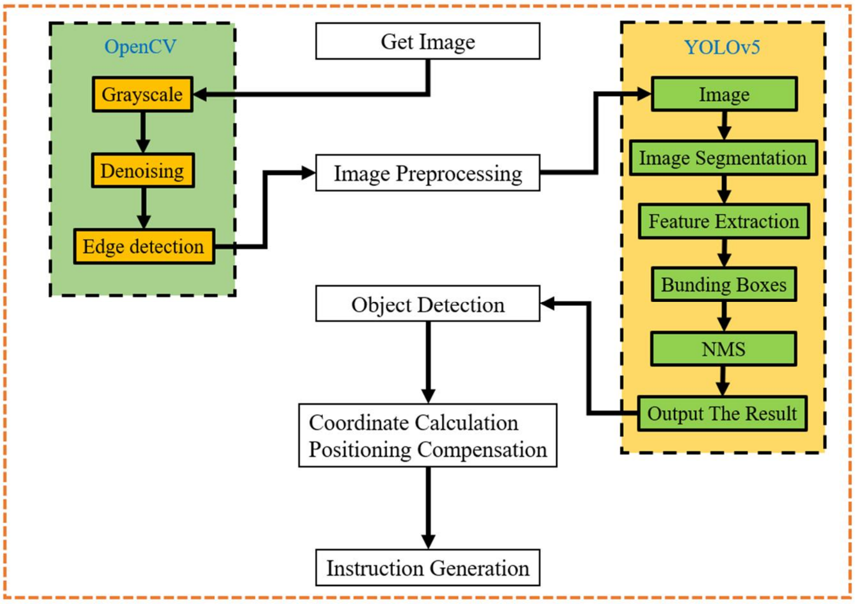

Figure 7. Image detection flow chart.

a) Image Acquisition:

Use a camera to capture images of the environment and transfer the data to the Raspberry Pi via USB or serial communication for processing. The camera is positioned in a carefully adjusted position to ensure that the field of view covers the area.

b) Image pre-processing:

Image pre-processing via OpenCV, including:

- Grayscale processing: Reduce the complexity of image data and improve processing efficiency.
- Denoising: Apply Gaussian blur or median filtering to reduce the impact of environmental interference on identification.
- Edge Detection: Enhance object contours with Canny edge detection to improve the accuracy of object detection.

c) Object detection: Input the preprocessed image data into the YOLO algorithm for object detection, run the following steps:

- Load the pretrained model (YOLOv5).
- Extract feature points in the image and generate candidate boxes.
- The final detection frame is screened by non-maximal suppression (NMS) to ensure that only the most relevant targets are labeled.

d) Coordinate Calculation and Positioning Compensation:

Based on the correction parameters of the camera, the image coordinates are converted to world coordinates to ensure that the robotic arm can be accurately positioned according to the detection results. A multi-point calibration method is used to eliminate the effects of lens distortion.

e) Command generation and run:

The Raspberry Pi generates control instructions based on the recognition results and transmits them to the servo motor and robotic arm controller through serial communication. The robotic arm completes the clamping or placing operation of the target object according to the instructions.

2.3.4. Object detection YOLO model training

In order to enable the intelligent robot to accurately identify workpieces, the YOLO (You Only Look Once) object detection algorithm was used in this study, and the model was trained and adjusted for industrial scenarios, as shown in **Figure 8**, and the specific process is as follows:

a) Data preparation
- Dataset collection: Collect high-resolution images covering the features of the workpiece, including different viewing angles, lighting conditions and background environment, to improve the generalization ability of the model.
- Data annotation: Use LabelImg to label images, including the position and category of the workpiece's rectangular box.

b) Model settings
- Select model version: This study chooses to use YOLOv5 as the core algorithm of object detection to train the model, which has the advantages of framework simplicity, which is easier to apply and develop, and the model can choose YOLOv5-L, YOLOv5-M, YOLOv5-S, etc., and the appropriate model size can be selected according to the development equipment.
- Model configuration: training and test data, artifact category name, total artifact category.

c) Model training
- Training Environment Settings: Use the Python environment to build the YOLO training framework.
- Execute training: Set training parameters, including image size, batch size, number of training rounds, etc., and execute commands to start training.
- Training Monitoring and Optimization: Monitor model loss, mAP, and other metrics in real time.

d) Model validation and testing
- Validate the dataset: Validate model performance using a test dataset, checking metrics including: mAP, Precision, Recall.
- Model Fine-tuning: Fine-tune the model based on the validation results, including optimizing the anchor point size or adjusting the input resolution to further improve detection accuracy.

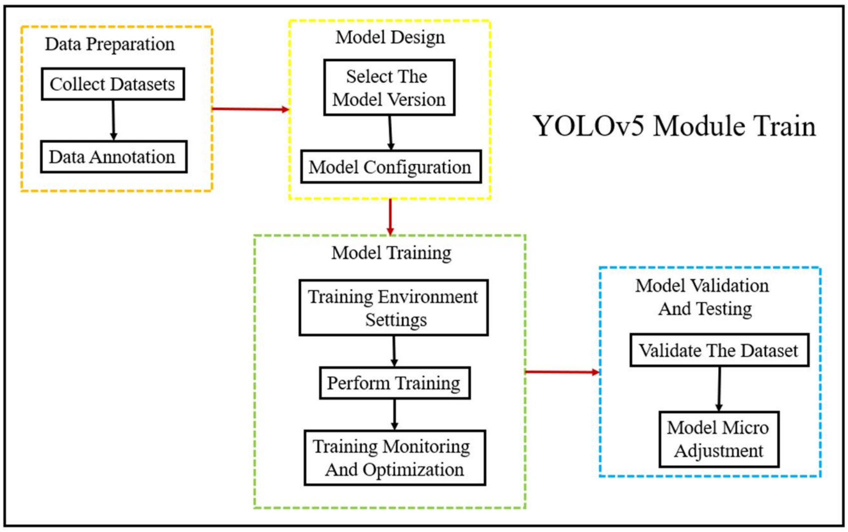

Figure 8. YOLOv5 model training.

3. Results

This study designed an intelligent robot system, illustrated in **Figure 9**. The system combines a self-propelled vehicle, a robotic arm, and advanced image recognition technology, aiming to assist traditional industries in transitioning towards industrial automation. The intelligent robot provides high mobility, allowing it to navigate various complex work environments, and it is equipped with a robotic arm capable of gripping items or performing other tasks, meeting diverse production line needs.

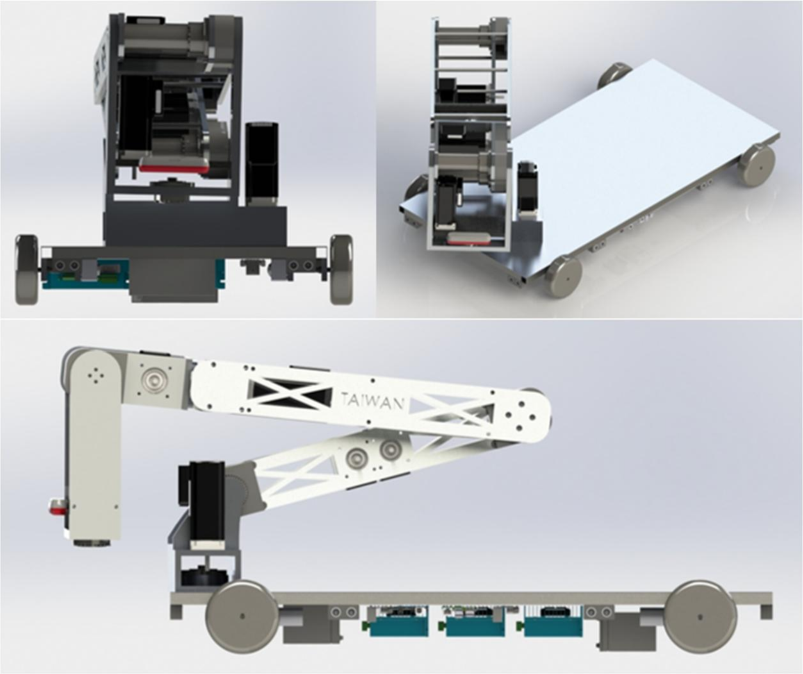

Figure 9. Intelligent robot design drawing.

Through rigorous theoretical analysis and experimental validation, the system demonstrated design reliability and practicality. Overall, the robot developed in this study is efficient and practical, capable of significantly improving productivity in small to medium-sized traditional industries and offering new possibilities for the future of intelligent manufacturing. Moreover, the system uses a Raspberry Pi as the controller, which is cost-effective compared to solutions from large enterprises. The design prioritizes commonly available components, reducing maintenance costs and further enhancing economic efficiency.

3.1. Expected performance of system functions

The intelligent robot has completed a number of technologies, and the expected results are shown in **Table 1**. In the operation, the system relies on a six-axis robotic arm with a camera to carry out real-time detection of the position of the workpiece and operations such as gripping and placement. This process ensures the stability and accuracy of the clamping process, reduces the risk of damage to the workpiece during operation, and improves the finishing efficiency of the production line. At the same time, the self-propelled vehicle system combines GPS, Lidar, ultrasonic and other sensors to collect environmental information, achieve autonomous navigation and be able to respond flexibly to the environment. In addition, in order to ensure the stability, continuity and cost of the system, the system uses two groups of Raspberry Pi to control the self-propelled vehicle and the robotic arm respectively, and through communication to achieve collaborative operation, through real-time transmission of task information, can complete complex processes collaboratively or interactively, shorten the production cycle and improve work efficiency.

Table 1. The expected effect of the intelligent robot.

Functional categories	Description of the function	Key technologies	What to expect
operations	Detection of position Complete gripping and placement	Six-axis robotic arm camera	Improve efficiency
Autonomous navigation Dynamic obstacle avoidance	Plan routes and avoid obstacles according to environmental routes	GPS, Lidar, Ultrasonic Sensors	Ensure transportation performance and safety in complex environments
Intelligent collaboration	Instant communication between the self-propelled vehicle and the robotic arm ensures the continuity of operations	Shumeipai embedded system and communication module	Improve the operation of the system.

The expected performance specifications of the intelligent robot in this study are shown in **Table 2**. In this study, the robotic arm has six degrees of freedom and the end of the arm is expected to lift a workpiece of about 5 kg, and the positioning accuracy is expected to reach ± 0.1 mm, and the camera can complete high-precision grasping work and other tasks, and in the conveying process, sensors are used to collect environmental data to navigate. The expected effect is that when the load reaches 45 kg, the workpiece can still be transported stably. The system uses Raspberry Pi to collaborate with UART, Wireless and other communication technologies to complete the goal of collaboration.

Table 2. Expected performance specifications of intelligent.

Project	**Specification**
Robotic arm degrees of freedom	6 DOF
robotic arm maximum payload	5 kg
precision	± 0.1 mm
Navigation	GPS, Lidar, Ultrasound
vehicle weight	30 kg
Carrying capacity	40 kg
Image recognition technology	YOLO (you Only Look Once)
Control system	Raspberry Pi
Methods of Communication	UART, Wireless
Energy supply	24 V 34 Ah Battery
Applicable environment	Temperature range 0 ℃ to 50 ℃

3.2. Structural design analysis

Figure 10 shows the stress analysis diagram of the load knot in the self-propelled vehicle used in this study. We used a square tube of ASTM A36 material, welded it and applied a 50~80 kg force on the surface to carry out the simulation analysis. The color distribution ranged from blue to red, indicating the limit value of the descent strength.

The subsidence strength of ASTM A36 steel is about 250 Mpa. When the stress is greater than this value, the material may be permanently deformed or fail. According to the analysis diagram, the maximum stress is about 167 MPa, which is lower than the subsidence strength of ASTM A36 steel. This means that when a force of 45KG is applied, this structure is safe and has not yet reached the yield point of structural stress. To ensure safety, the maximum load should consider the factor of safety. If the safety factor is set at 1.5, the maximum allowable stress is as shown in Equation (1):

$$\sigma_{max} = \frac{subsidence\ strength}{Safety} = \frac{250MPa}{1.5} \approx 167MPa \tag{1}$$

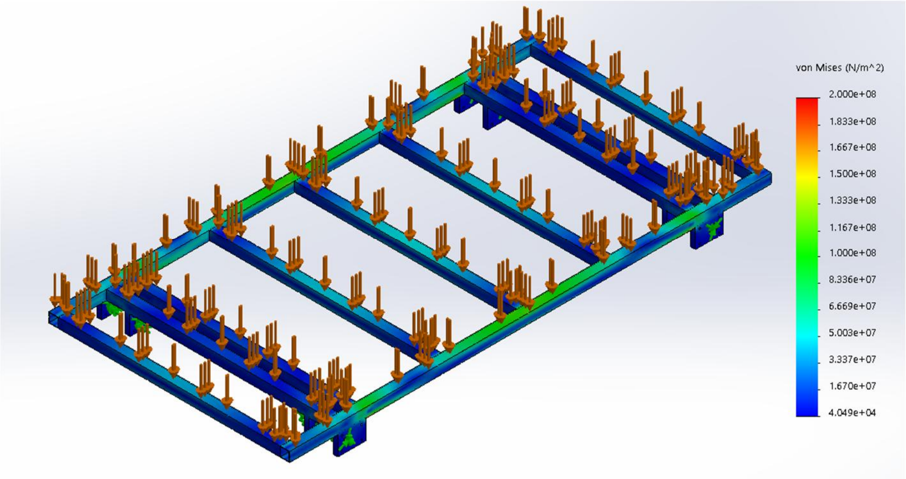

Figure 10. The static stress distribution diagram of 45 kg.

When a simulated force of 70 kg is applied, as shown in **Figure 11**, the maximum stress distribution value reaches approximately 183 MPa. This exceeds the allowable safety factor range and approaches the critical value of the material subsidence strength, indicating the potential for permanent deformation or failure under this load. To ensure the stability and reliability of the structure in long-term use, it is recommended that the load limit be set to less than 50 kg.

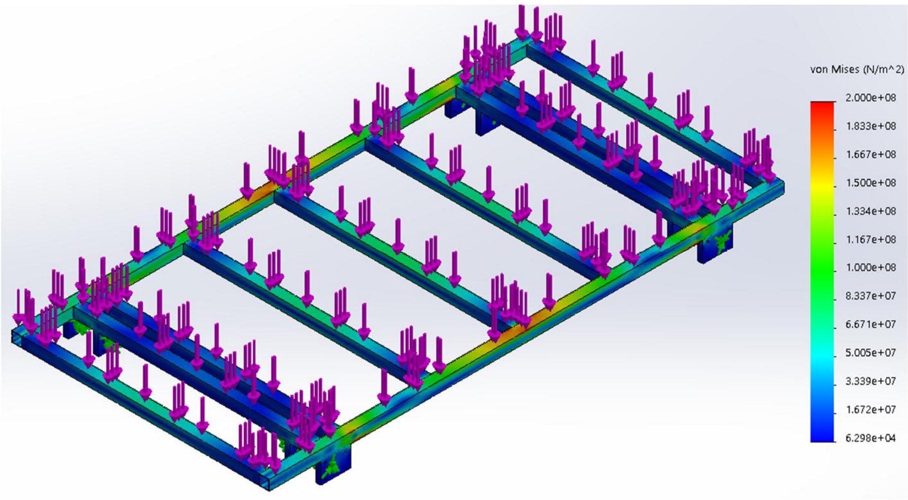

Figure 11. The application of 70 kg static stress analysis diagram.

When a force of about 80 kg is applied, as shown in **Figure 12**, the stress distribution has been clearly red area, which means that the force generated at this time has exceeded the limit that the material can bear, so it can be seen that the chassis structure of the self-propelled vehicle in this study can withstand the force of less than 80 kg when it exceeds the safety factor range.

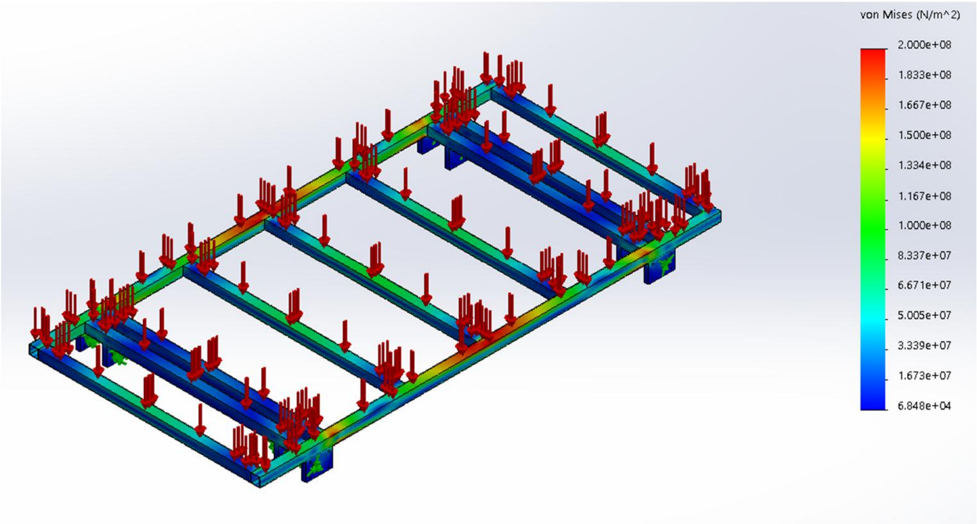

Figure 12. The application of 80 kg static stress analysis diagram.

3.3. Model training results

This study initially utilized approximately 50 images of workpieces in diverse environments for training. Post-optimization, the performance indicators are shown in **Table 3**. Through evaluation with the test set, the model swiftly determined the workpiece's location in the environment. Furthermore, the YOLOv5 model was deployed on a Raspberry Pi. Given the option to select model files, we chose YOLOv5-s. The model achieved real-time operational performance in an embedded system. After multiple rounds of testing, this study developed a target detection system for workpieces. This system assists intelligent robots in accomplishing automated tasks, thereby enhancing the manufacturing and production capabilities of traditional factories.

Table 3. Performance metrics for image recognition models.

Index	**Number**
mAP	94%
Precision	93%
Recall	91%
Inference frame	18 Fps

4. Conclusion

This study addresses the automation needs of small and medium-sized traditional manufacturing plants by proposing an intelligent robot tailored for industrial automation. The design integrates mechanical engineering, circuitry design, and software development. The robotic system consists of a six-degree-of-freedom robotic arm and a mobile platform to ensure high flexibility and stability. Additionally, the system incorporates cameras and YOLO (You Only Look Once) technology for accurate object recognition. The circuitry design, based on Raspberry Pi and Arduino control architecture, effectively integrates various sensors and communication modules, ensuring efficient collaborative operations. The software design leverages OpenCV and YOLO algorithms, enhancing image recognition accuracy and the ability to handle complex industrial scenarios.

Although a prototype of the intelligent robot was not physically constructed, the theoretical design verified the system's feasibility and potential benefits. The proposed system can shorten production cycles, improve operational precision, and adapt to diverse industrial environments. Future research should focus on actual prototype development and testing to further validate the system's stability and performance. Additionally, optimizing computational efficiency and reliability can further enhance the system's capability to handle complex industrial scenarios.

Author contributions: Conceptualization, YHC and SYP; methodology, YHC and SYP; software, SYP; validation, YHC and SYP; formal analysis, YHC and SYP; investigation, YHC and SYP; resources, YHC; data curation, YHC and SYP; writing—original draft preparation, YHC and SYP; writing—review and editing, YHC; visualization, YHC and SYP; supervision, YHC; project administration, YHC and SYP; funding acquisition, YHC. All authors have read and agreed to the published version of the manuscript.

Conflict of interest: The authors declare no conflict of interest.

References

1. Mabkhot MM, Al-Ahmari AM, Salah B, et al. Requirements of the Smart Factory System: A Survey and Perspective. Machines. 2018; 6(2): 23. doi: 10.3390/machines6020023
2. Kalsoom T, Ramzan N, Ahmed S, et al. Advances in Sensor Technologies in the Era of Smart Factory and Industry 4.0. Sensors. 2020; 20(23): 6783. doi: 10.3390/s20236783
3. Zinn M, Roth B, Khatib O, et al. A New Actuation Approach for Human Friendly Robot Design. The International Journal of Robotics Research. 2004; 23(4-5): 379-398. doi: 10.1177/0278364904042193
4. Zhang Y, Guo Z, Wu J, et al. Real-Time Vehicle Detection Based on Improved YOLO v5. Sustainability. 2022; 14(19): 12274. doi: 10.3390/su141912274
5. Mahdi E, Eltai E, Alabtah FG, et al. Mechanical Characterization of AA 6061-T6 MIG Welded Aluminum Alloys Using a Robotic Arm. Key Engineering Materials. 2022; 913: 271-278. doi: 10.4028/p-rhrr3n
6. Garzón Quiroz MQ. Inductive Machine Learning with Image Processing for Objects Detection of a Robotic Arm with Raspberry PI. In Technology Trends; 2019. doi: 10.1007/978-3-030-05532-5_45
7. Zhao Y, Liu X, Cao J, et al. A calibration and compensation method for an industrial robot with high accuracy harmonic reducers. Science China Technological Sciences. 2024; 67(3): 725-739. doi: 10.1007/s11431-023-2526-2
8. Cheng HT, Sun Z, Zhang P. Real-Time Imitative Robotic Arm Control for Home Robot Applications. In: Proceedings of the IEEE International Conference on Robotics and Automation; 2011.
9. Lu L, Liu B, Mao E, et al. Design and Optimization of High Ground Clearance Self-Propelled Sprayer Chassis Frame. Agriculture. 2023; 13(2): 233. doi: 10.3390/agriculture13020233
10. Srivastava S. Raspberry Pi Controlled Robotic Arm. International Journal of Research. 2015.
11. Kadir WMHW, Samin RE, Ibrahim BSK. Internet Controlled Robotic Arm. Procedia Engineering. 2012; 41: 1065-1071. doi: 10.1016/j.proeng.2012.07.284
12. Calvo I, Gil-García J, Recio I, et al. Building IoT Applications with Raspberry Pi and Low Power IQRF Communication Modules. Electronics. 2016; 5(3): 54. doi: 10.3390/electronics5030054
13. Viswanatha V, Chandana RK, & Ramachandra AC. Real-Time Object Detection System with YOLO and CNN Models: A Review. arXiv preprint arXiv:2208.00773; 2022.
14. Hu H, Yu L, Wo Tsui P, et al. Internet-based robotic systems for teleoperation. Assembly Automation. 2001; 21(2): 143-152. doi: 10.1108/01445150110388513
15. Somarathna RN. Path Planning and Obstacle Avoidance Scheme for Autonomous Robots using Raspberry Pi. arXiv preprint arXiv:2012.10863; 2020.
16. Redmon J, Divvala S, Girshick R, et al. You Only Look Once: Unified, Real-Time Object Detection. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016.
17. Wang Z, Li C, Xu H, & Zhu X. Mamba YOLO: SSMs-Based YOLO for Object Detection. arXiv preprint arXiv:2406.05835; 2024.
18. Terven J, Córdova-Esparza DM, Romero-González JA. A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. Machine Learning and Knowledge Extraction. 2023; 5(4): 1680-1716. doi: 10.3390/make5040083
19. Zhong Y, Wang J, Peng J, et al. Anchor Box Optimization for Object Detection. 2020 IEEE Winter Conference on Applications of Computer Vision (WACV). Published online March 2020: 1275-1283. doi: 10.1109/wacv45572.2020.9093498
20. Deng C, Wang M, Liu L, Liu Y. Extended feature pyramid network for small object detection. arXiv preprint arXiv:2004.00950; 2020.