

Development of an AI and Webserver-integrated Smart Automated Storage and Retrieval System

Quang-Thien Nguyen¹, Thien-Bao Truong², Tan-Huy Tran^{3,*}, Tan-Loc Nguyen⁴, Ngoc-Son Vo⁵, Nguyen-Khang Bui⁶,
Van-Dong-Hai Nguyen⁷, Thanh-An Cao⁸, Thi-Ngoc-Thao Nguyen⁹, Thi-Hong-Lam Le¹⁰
^{1, 2, 3, 4, 5, 6, 7, 8, 9, 10} Ho Chi Minh City University of Technology and Engineering (HCM-UTE), Ho Chi Minh City (HCMC), Vietnam
Email: ¹ 23161196@student.hcmute.edu.vn, ² 21146428@student.hcmute.edu.vn, ³ 21146469@student.hcmute.edu.vn,
⁴ 21146125@student.hcmute.edu.vn, ⁵ 24151256@student.hcmute.edu.vn, ⁶ 24151170@student.hcmute.edu.vn,
⁷ hainvd@hcmute.edu.vn, ⁸ 22161044@student.hcmute.edu.vn, ⁹ thaontn@hcmute.edu.vn, ¹⁰ lamlth@hcmute.edu.vn
*Corresponding Author

Abstract—In recent years, Automated Storage and Retrieval Systems (AS/RS) and their development have been a notable trend of modern warehouse management by automating the sequential and precise processes of storing, sorting, and retrieving goods. Driven by the convergence of mechatronic systems, Industrial Internet of Things (IIoT), Artificial Intelligence (AI), cloud storage, and edge-based management systems, the potential and practical benefits of AS/RS can be significantly amplified when effectively combined with these trends. In this field, although some works are presented, they often lack specialization for the Vietnamese industrial environment and sustainability. Therefore, this research presents the development of an intelligent AS/RS, incorporating AI-based label processing and webserver-based control to enhance warehouse management efficiency. Experimental evaluations demonstrate that the system achieves high reliability in product classification and storage tasks, providing a scalable solution for modern smart logistics with real-time data synchronization capabilities via a Node-RED web server.

Keywords—Automated Storage and Retrieval System; YOLOv8; Vintern-1B; PLC S7-1200; IIoT; Webserver

I. INTRODUCTION

In the context of the e-commerce boom and the rise of complex supply chains, requirements for response times have exposed the inherent limitations of manual warehousing systems [1]. These constraints include inefficiencies in physical space allocation, high error rates due to human intervention, increasing labor costs, and poor adaptability to market conditions. Consequently, there is a transition from traditional warehouses to modern urban storage systems to minimize transportation costs and further address traffic and environmental issues [2]. Within this landscape, Automated Storage and Retrieval Systems (AS/RS) are redefining the entire structure of modern warehouse management [3]. By leveraging technologies such as robotics and control software, AS/RS optimizes storage space by utilizing the full warehouse height, eliminating unnecessary gaps between racking units, and automating the sequential and precise storage, sorting, and retrieval of goods [4]. In an AS/RS, the retrieval process is computer-managed, including decisions on storage locations, movement from input to output, retrieval sequencing, order execution, and inventory status updates, allowing it to operate independently or integrate into workstations [5].

Furthermore, aligned with the convergence of Artificial Intelligence and the Industrial Internet of Things (IIoT), AS/RS design trends are shifting to meet more complex

corporate demands. Notably, applications of computer vision, especially CNNs and web servers, are emerging to overcome AS/RS drawbacks in harsh environments, improving operational efficiency, management, and system utilization. Prominent studies such as [5]-[10] have confirmed the vital role of integrating AI and IIoT into AS/RS. These works investigate the use of semantic segmentation networks like YOLOv8 for identifying and counting cargo boxes [8], monitoring systems via web servers [6], or cloud-based programmable control to enhance system flexibility [7]. Additionally, research aimed at improving industrial AI performance, such as class-specific dataset splitting for YOLOv8 [11] and enhanced Optical Character Recognition (OCR) through information localization, has laid the foundation for subsequent applications.

However, current works still exhibit weaknesses, particularly a lack of specialization for the Vietnamese industrial environment. QR code recognition systems prove inadequate in harsh conditions where cargo collisions and friction lead to torn or blurred code may cause significant losses for warehouses [12]. To address this, various studies have utilized AI to process product character labels when QR codes cannot be extracted. A common improvement involves combining the YOLO series (e.g., YOLOv5, YOLOv8) for high-speed label localization with an OCR network like PaddleOCR [13] for character recognition, as demonstrated in [12], [14]. Although this addresses the QR issue, traditional OCR networks like PaddleOCR are not optimized for the Vietnamese language and lack semantic awareness, requiring rigid regular expressions (regex), which are error-prone and inflexible. Furthermore, YOLO models pre-trained on the COCO dataset [15] only recognize common objects and are not specialized for industrial environments. On this basis, this research proposes and successfully implements an AI-integrated AS/RS model deployed on a web server tailored for not only Vietnam, but also maintains high adaptability for various industrial environments. The model employs YOLOv8 to detect and localize QR codes or labels via camera and transmits this data to Vintern-1B [16], a 1B-parameter multimodal large language model (MLLM) with robust Vietnamese OCR capabilities suitable for resource-constrained devices. Subsequently, a webserver was developed on the Node.js platform, utilizing Socket.IO for real-time data transmission and the Nodes7 library for direct TCP/IP communication with a Siemens S7-1200 PLC. This

web server serves not only as a user-friendly Human-Machine Interface (HMI) on browsers but also as a processing distribution center: receiving JSON results from the AI model, logging them into a database, and outputting trigger signals to activate PLC I/O pins automatically.

The structure of this paper is as follows: Section. II presents the hardware fabrication principles and software design; Section III describes the experimental evaluation and results; and Section. IV provides the conclusion and future research directions.

II. METHODOLOGY

A. Hardware Designing and Control Algorithm

The mechanical structure is developed based on a 3-axis Cartesian coordinate robot. Each axis is driven by a NEMA 17 stepper motor, ensuring high torque and precision during the transportation of goods into the Selective racking system. The control logic is executed on a Siemens S7-1200 PLC (CPU 1214C DC/DC/DC). Motion trajectories are optimized using a linear interpolation algorithm when delivering cargo into the storage cells. The general system is described in Fig. 1.

The physical system model (Fig. 2), consists of three independent motion axes: the X-axis moves along the aisle to cover the warehouse length (Fig. 3), the Z-axis performs lifting and lowering of the platform to access various rack levels (Fig. 4 and Fig. 5), and the Y-axis executes the reach operation to place or retrieve goods from the storage cells. The single selective racking system is divided into defined coordinate grids, allowing the robot to directly access any cargo position without moving surrounding items. To implement intelligent functions, a high-resolution camera is installed at the infeed conveyor area to collect product images for identification data processing by AI models (YOLOv8 and Vintern-1B).

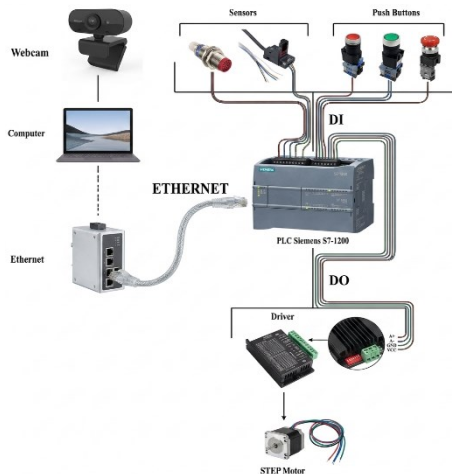


Fig. 1. General hardware diagram

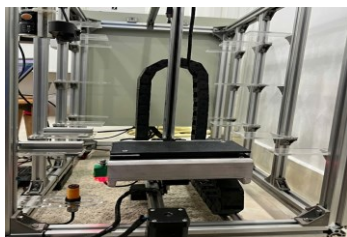


Fig. 2. Mechanical model

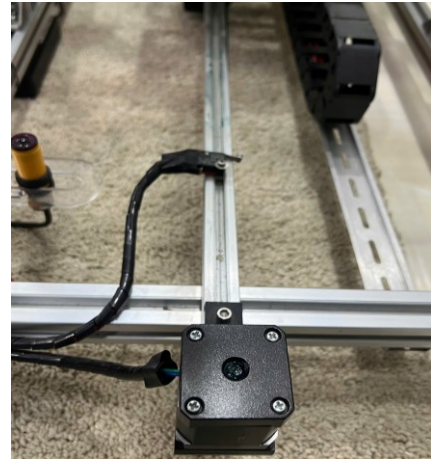


Fig. 3. X-axis movement mechanism

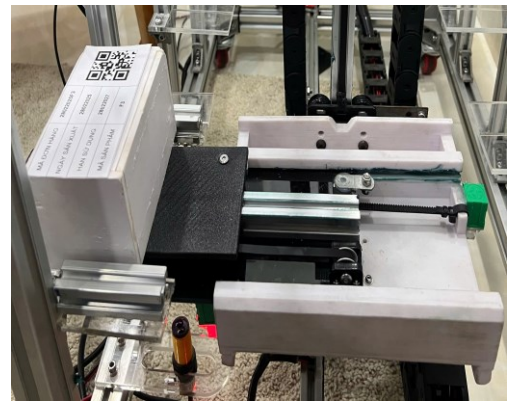


Fig. 4. Lifting mechanism



Fig. 5. Z-axis movement mechanism

The system's control functions are divided into two main operations: storage and retrieval. During the storage mode, goods are scanned and must be transported to a specific point equipped with an identification sensor. If the sensor fails to detect any cargo, the robot will stop at its last working position and await further instructions. For instance, if the robot completes delivering an item to slot 5 and no product signal is detected by the identification sensor, the robot will remain stationary at slot 5 until a new command is issued. These automatic storage processes are described in Fig. 6 and Fig. 7.

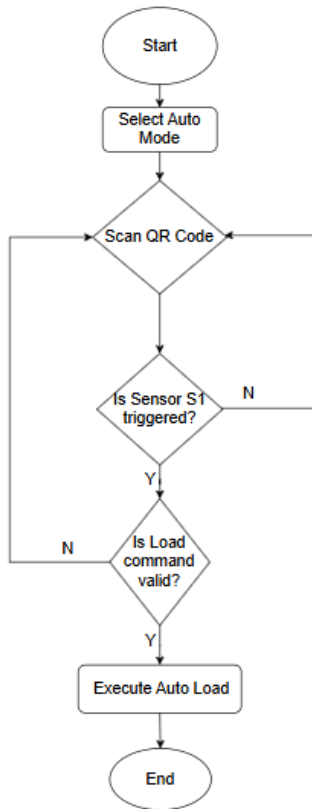


Fig. 6. Flowchart of the automatic operation mode

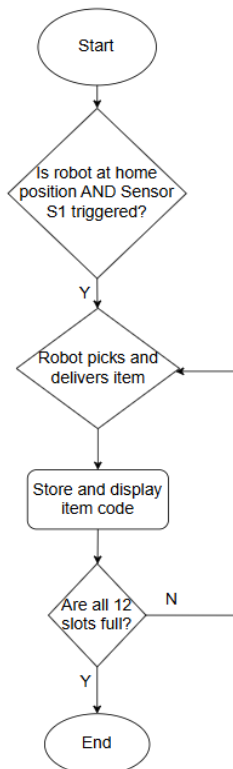


Fig. 7. Loading process state diagram

In retrieval mode, the robot moves to the designated slot, retrieves the item, and transports it back to the staging area. Upon completion of the retrieval task, the robot returns to the Home position and remains on standby for the next command to resume operations. The process is demonstrated in Fig. 8.

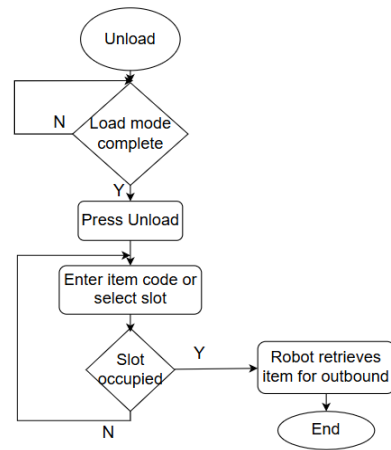


Fig. 8. Unloading process diagram

B. AI-Powered Vision System

The vision pipeline is established to replace manual inspection processes. The workflow is executed through two primary stages (Fig. 9):

- Object Detection: The YOLOv8 model is employed to localize product labels or QR codes and output this information to the PLC or the Vintern-1B model.
- Quality Inspection: The Vintern-1B model receives the image data to perform OCR when QR codes are corrupted. This includes Optical Character Recognition (OCR) for label verification and visual defect detection. The data extracted via OCR is utilized as a substitute for the missing or damaged QR code information.

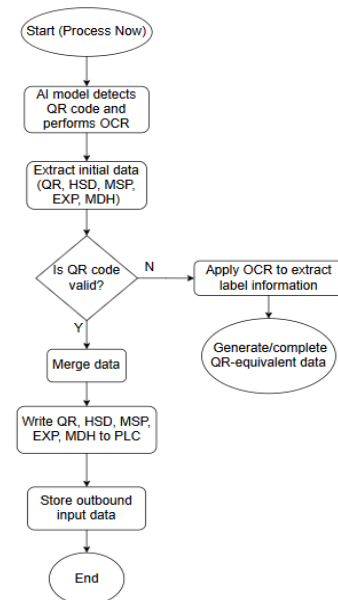


Fig. 9. AI processing flowchart

C. Web-Based Monitoring System

The system integrates a WebServer built on the Node-RED platform, enabling remote monitoring and control of the warehouse status. Data, including product classification, storage slot locations, and storage timestamps, is continuously updated in real-time. Fig. 10 to Fig. 12 show basic working interfaces of the system

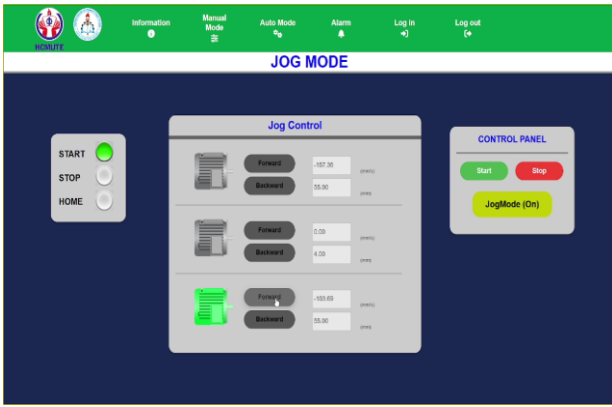


Fig. 10. Jog mode interface

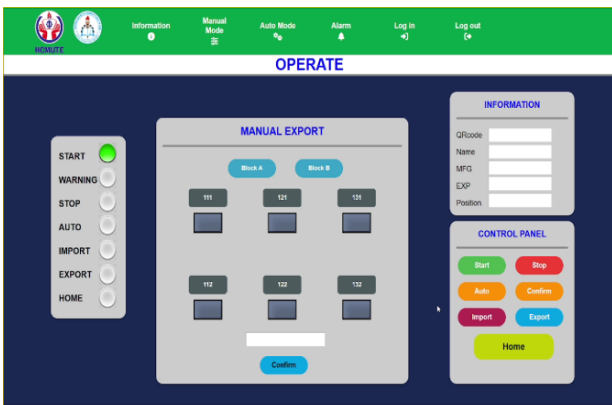


Fig. 11. Operating mode interface

- Notification Log: This area displays real-time operational logs, capturing essential events such as AI inference results, PLC connectivity status, and system-wide alerts. Every entry is automatically timestamped for better operational clarity.

Users can interact via the webserver to execute automated storage tasks and manual or scan-based retrieval operations. For automated storage, the user places the cargo at the storage position and initiates the QR code (or OCR) scanning process. After that, the result will be displayed on the screen as shown in Fig. 13.

For retrieving, users can choose storage cells (green) to mark it (yellow) or use images/QR, working interfaces are shown in Fig. 14 to Fig. 16.

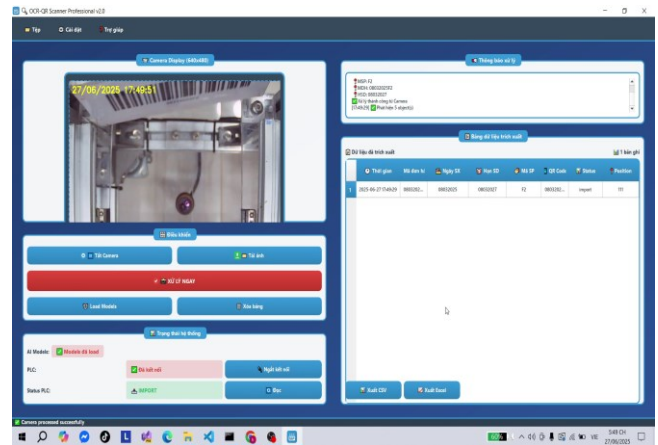


Fig. 13. Storing completed, storing good's information into database

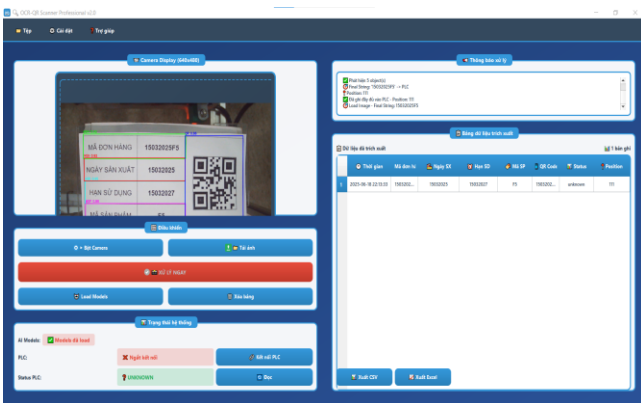


Fig. 12. AI model interface



Fig. 14. Retrieving manually

Interface Components in Fig. 13:

- Camera Frame: Positioned within a dedicated container in the left column, the camera feed is integrated into a specialized frame with precise dimensions of 660x520px. This includes a 10px padding on all sides to maintain a clear and stable video display at a standard 640x480px resolution.
- Control Panel: Located in the left column, this section utilizes a flexible grid layout to organize interactive command buttons. These include functions for Camera Power (On/Off), Image Upload, Manual Processing, Model Loading, and Table Reset.
- Data Table: A dynamic web table comprising eight specialized columns: Timestamp, Order ID, Production Date, Expiry Date, Product ID, QR Code, Status, and Position.

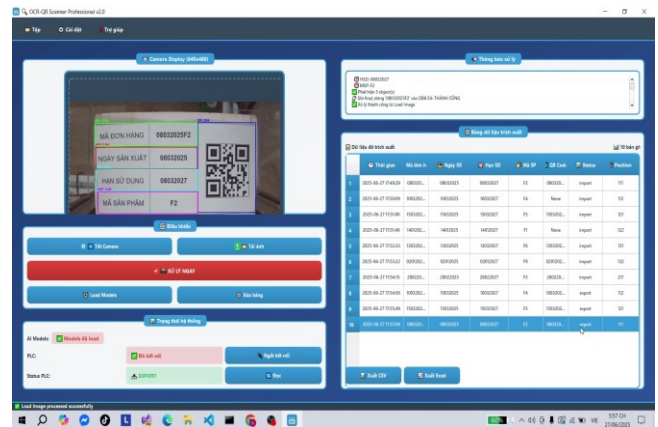


Fig. 15. Retrieving with images

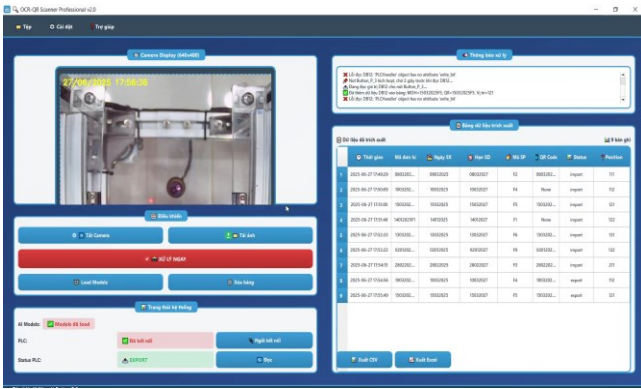


Fig. 16. Storing and Retrieving information

III. EXPERIMENTAL RESULTS AND DISCUSSION

Based on the aforementioned design, the research team conducted experiments to evaluate the improvement and effectiveness of the experimental simulation. The hardware configuration utilized includes a Siemens S7-1200 1214 DC/DC/DC PLC, NEMA 17 stepper motors (size 42), and TB6600 drivers, with the wiring diagram as illustrated in the figure. The sensing system comprises E3F-DS30P1 obstacle detection sensors and FOTEK PL-05P PNP inductive proximity sensors. The system operates fully automatically through the integration of the Webserver, PLC, robot, and AI-powered camera. The AI model is responsible for product identification via QR codes or OCR, ensuring high accuracy in warehouse management.

A. AI Performance Evaluation

Following the training and validation processes, the YOLOv8 model demonstrated high precision in identifying a diverse range of product classes.

The training process as shown in Fig. 17 indicates high learning efficiency without overfitting, as both training and validation losses (box, class, and dfl loss) consistently converged toward zero over 100 epochs. Key performance metrics reached stability, with precision maintaining approximately 0.8, while recall improved from 0.7 to 0.9. Notably, the model achieved a mAP50 of 0.9, and a mAP50-95 ranging between 0.6 and 0.8, demonstrating robust detection capabilities while suggesting further optimization potential for higher IoU thresholds.

The bar chart in Fig. 18 illustrates the total number of scans, successful QR code scans, and successful OCR scans for each product code from F1 to F9. To ensure experimental consistency, each product code (F1–F9) was subjected to 40 scans, this sample size ensures the reliability of the system in long-term operation, while keeping the hardware from being overloaded in experimental environment. Most products (F2, F3, and F5–F7) achieved a 100% QR success rate (40/40). However, F1, F4, F8, and F9 exhibited lower rates (36/40, 37/40, 39/40, and 38/40, respectively), reflecting real-world scenarios of corrupted QR codes. The 'OCR scan count' reached 40 successful attempts for nearly all products, even when the QR code was successfully read at 100%. This confirms that OCR operates in parallel with QR scanning as a dual verification and backup mechanism. In cases of QR failure (F1, F4, and F8), OCR successfully restoring the required information.

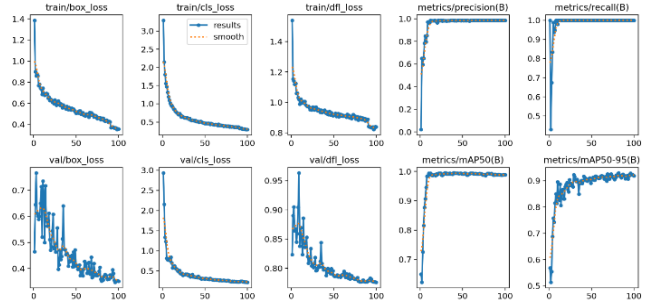


Fig. 17. Training and validating charts

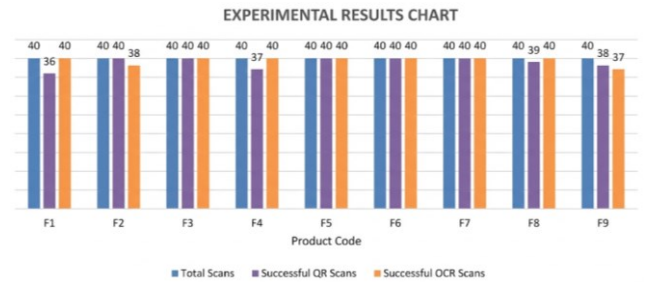


Fig. 18. Experimental results chart

Fig. 19 illustrates the success rates (in percentage) for both QR code scanning and OCR for each product code from F1 to F9.

- QR Code Success Rate: The QR scanning performance reached 100% for the majority of products, with slightly lower rates for F1 (90%), F4 (93%), F8 (98%), and F9 (95%).
- OCR Success Rate: The OCR accuracy reached 100% for most items, with F2 at 95% and F9 at 92.5%, this may be caused by the light condition which affect OCR ability while QR scans remained robust due to its high light-resistance capability. These results confirm the system's overall ability to restore and maintain high data integrity through OCR, even when the initial QR data is suboptimal.

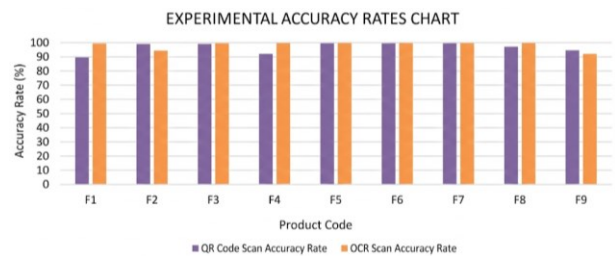


Fig. 19. Comparison between the QR and OCR scanning success rates

B. System Operational Efficiency

The research team measured the total cycle time for a single storage/retrieval operation, encompassing both AI processing and mechanical robot travel time. The results indicate an average cycle time of 17 seconds, with a 100% successful mechanical positioning rate.

C. Webserver Interface

The web server is designed to ensure ease of operation, intuitive monitoring, efficient risk handling, and overall system safety. Within this system, the web server supports user account authentication with persistent login credential

storage (as illustrated). It provides a comprehensive and visually intuitive interface comprising the following components: a logging site (Fig. 20); an operating mode panel supporting both automatic and manual control modes (Fig. 21); an AI model interface featuring dedicated control buttons, real-time operational status indicators and live visual feedback streamed directly from the camera module (Fig. 22) and a data logging table enabling full traceability of operational records, with export functionality to Excel format for systematic data management and reporting purposes (Fig. 23).

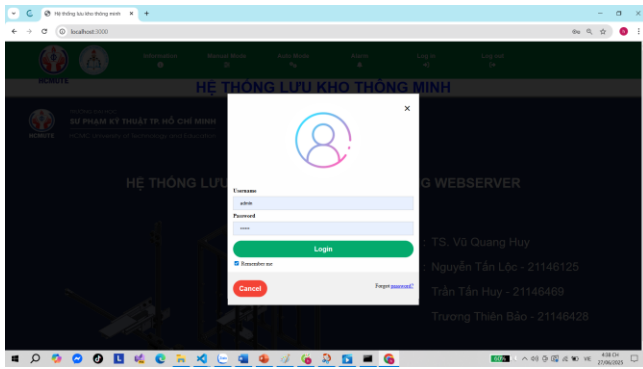


Fig. 20. Logging interface

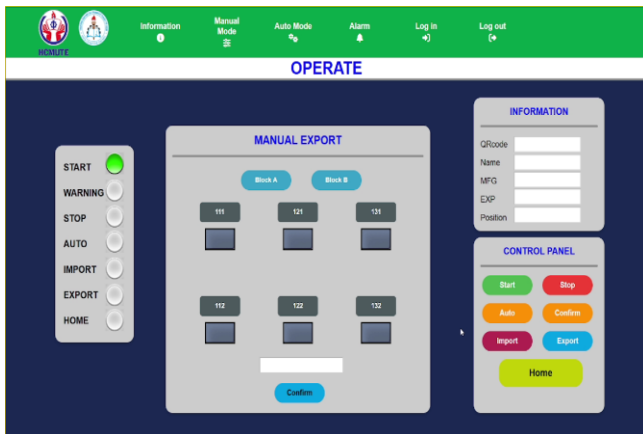


Fig. 21. Automatic operational mode interface

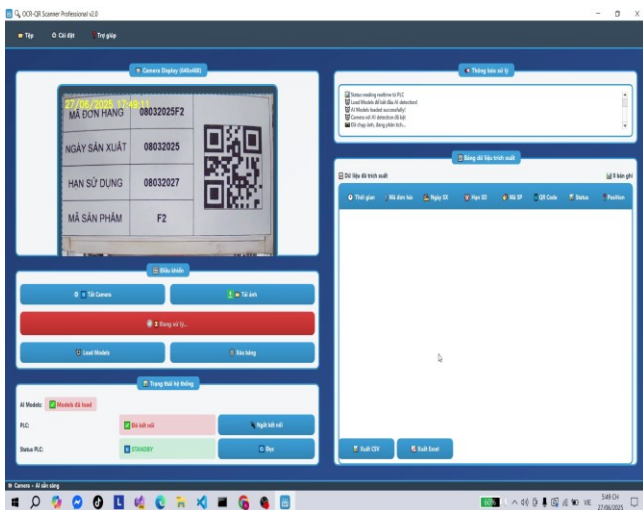


Fig. 22. AI-based system interface for QR scanning and control

	Thời gian	Đơn	Ngày SX	Hạn SD	Mã SP	QR Cc	Status
1	2025-06-28 20:42:13	130...	13032025	13032027	F6	130...	export
2	2025-06-28 20:42:09	130...	13032025	13032027	F6	130...	export
3	2025-06-28 20:42:06	130...	13032025	13032027	F6	130...	export
4	2025-06-28 20:41:29	100...	10032025	10032027	F4		import
5	2025-06-28 20:40:52	150...	15032025	15032027	F5	150...	import
6	2025-06-28 20:40:18	020...	02012025	Không có dữ liệu phù hợp nào từ hình ảnh.	F9	020...	import
7	2025-06-28 20:39:51	140...	20240101	14012027	F1		import
8	2025-06-28 20:39:16	080...	08032025	08032027	F2	080...	import
9	2025-06-28 20:38:53	130...	13032025	13032027	F6	130...	import

Fig. 23. Data storage and retrieval information interface

IV. CONCLUSION

In modern industrialization, optimizing storage space and automating warehouse management processes have become crucial to enhance corporate competitiveness. This research addressed the core limitations of traditional warehousing systems, specifically the lack of automated quality control and the insufficient flexibility in handling damaged product labels. The implemented solution involved developing an intelligent AS/RS that synchronously integrates Siemens S7-1200 PLC logic control, advanced AI-based computer vision, and real-time monitoring via a Webserver. A key breakthrough of this study lies in the integration of the YOLOv8 model for object identification and the Vintern-1B multimodal large language model (MLLM) for OCR data restoration in cases of QR code failure.

Experimental results have confirmed the effectiveness and reliability of the proposed model under practical operating conditions. The system achieved an average storage/retrieval cycle time of 17 seconds, successfully meeting the initial productivity target of under 20 seconds. Regarding AI performance, the detection model reached a mAP@0.5 precision of 0.992, while OCR-based data verification achieved an accuracy range of 92.5% to 100%, even with corrupted QR labels. The system maintained a 100% successful positioning rate, with inventory data consistently synchronized to the Node-RED monitoring interface via TCP/IP communication.

Despite these positive outcomes, certain limitations remain, such as the AI processing speed's dependence on computer hardware configuration and network latency occasionally affecting real-time data updates on the Webserver. Furthermore, the current model primarily focuses on uniform cargo sizes and labels within a single Selective racking system. Future research will focus on optimizing AI model architectures to reduce inference time, expanding recognition capabilities for more complex packaging types, and integrating advanced identification protocols such as RFID. Developing path optimization algorithms for robots in multi-aisle warehouse environments also remains a key objective to move towards a comprehensive and flexible intelligent warehouse management system for large-scale industrial applications.

ACKNOWLEDGMENT

This research was funded by Ho Chi Minh City University of Technology and Engineering, Vietnam, under grant No. SV2026-267. We want to give thanks to Ph.D. Quang-Huy Vu (HCM-UTE) due to his supervision for this project. We, authors, are grateful for these supports. Link of operation of system is: <https://youtu.be/5uLDbmLTdiE?si=iPUxwkEgU3hs7JnG>.

REFERENCES

- [1] A. Edouard, Y. Sallez, V. Fortineau, S. Lamouri, and A. Berger, "Automated storage and retrieval systems: An attractive solution for an urban warehouse's sustainable development," *Sustainability (Switzerland)*, vol. 14, no. 15, p. 9518, 2022, <https://doi.org/10.3390/su14159518>.
- [2] S. Kang, "Why do warehouses decentralize more in certain metropolitan areas?," *Journal of Transport Geography*, vol. 88, p. 102330, 2020, <https://doi.org/10.1016/j.jtrangeo.2018.10.005>.
- [3] H. Majeed, A. T. Rashid, and K. A. Mohamad, "The Automatic Storage and Retrieval System: An Overview," *International Journal of Computer Applications*, vol. 177, no. 16, pp. 36–43, Nov. 2019, <https://doi.org/10.5120/ijca2019919603>.
- [4] H. Wang, Y. Wei, and H. Yan, "Automatic single table storage structure selection for hybrid workload," *Knowledge and Information Systems*, vol. 65, no. 11, pp. 4713–4739, 2023, <https://doi.org/10.1007/s10115-023-01913-7>.
- [5] F. H. Mohammad Khasasi, A. M. Abdul, and Z. Mohd Yusof, "Development of an Automated Storage and Retrieval System in dynamic industrial environment," in *2015 International Conference on BioSignal Analysis, Processing and Systems, ICBAPS 2015*, IEEE, pp. 57–60, 2015, <https://doi.org/10.1109/ICBAPS.2015.7292218>.
- [6] W. Sánchez Ocaña, B. G. Jácome, S. A. Cevallos, R. X. S. Paredes, and E. S. Jácome, "Design and construction of an automated pneumatic storage/retrieval system (AS/RS) by means of product recognition with QR codes," *International Review of Automatic Control*, vol. 14, no. 4, pp. 191–200, 2021, <https://doi.org/10.15866/ireaco.v14i4.20822>.
- [7] . K. M. Lee, Y. Lv, K. K. H. Ng, W. Ho, and K. L. Choy, "Design and application of internet of things-based warehouse management system for smart logistics," *International Journal of Production Research*, vol. 56, no. 8, pp. 2753–2768, 2018, <https://doi.org/10.1080/00207543.2017.1394592>.
- [8] S. Thaneesan and J. A. K. S. Jayasinghe, "YOLOv8 powered solutions for box identification in warehouses," *International Journal of Innovative Science and Research Technology (IJISRT)*, pp. 1560–1565, 2024, <https://doi.org/10.38124/ijisrt/ijisrt24sep1017>.
- [9] L. Lagsaiar, I. Shahrou, A. Aljer, and A. Soulhi, "Modular software architecture for local smart building servers," *Sensors*, vol. 21, no. 17, p. 5810, 2021, <https://doi.org/10.3390/s21175810>.
- [10] F. Alsulami and N. Z. Jhanjhi, "Deep learning framework for barcode localization and decoding using simulated UAV imagery," *Scientific Reports*, vol. 16, no. 1, p. 399, 2026, <https://doi.org/10.1038/s41598-025-29720-w>.
- [11] C. Tadjine, A. Ouafi, A. Taleb-Ahmed, and Y. El Hillali, "Class-Specific Dataset Splitting for YOLOv8: Improving Real-Time Performance in NVIDIA Jetson Nano for Faster Autonomous Forklifts," in *International Conference on Pattern Recognition Applications and Methods*, SCITEPRESS - Science and Technology Publications, pp. 788–793, 2025, <https://doi.org/10.5220/0013308700003905>.
- [12] K. Cui, Q. Xu, Y. Ding, J. Mei, Y. He, and H. Liu, "Optical Character Recognition Method Based on YOLO Positioning and Intersection Ratio Filtering," *Symmetry*, vol. 17, no. 8, p. 1198, 2025, <https://doi.org/10.3390/sym17081198>.
- [13] L. Hamdi, A. Tamasna, P. Boisson, and T. Paquet, "PILOT: A promptable interleaved layout-aware OCR transformer," *International Journal on Document Analysis and Recognition (IJ DAR)*, 2026, <https://doi.org/10.1007/s10032-026-00590-w>.
- [14] J. Yin, "An automatic extraction and classification method of electronic component tags combining OCR and object detection," in *International Conference on Mechatronics and Artificial Intelligence (MAI 2025)*, M.-C. Su and U. Gayh, Eds., SPIE, p. 58, 2025, <https://doi.org/10.1117/12.3086320>.
- [15] T. Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, pp. 740–755, 2014, https://doi.org/10.1007/978-3-319-10602-1_48.
- [16] K. T. Doan *et al.*, "Vintern-1B: An Efficient Multimodal Large Language Model for Vietnamese." 2024, <https://doi.org/10.48550/arXiv.2408.12480>.