Jurnal Komputer

Vol. 1, No. 1, October 2026

SAH PUBLISHER

Parkur: Digital Twin Parking System with Nearest-Slot Allocation and Edge AI-Based Occupancy Monitoring

Brieva Ommr[1*], Muhammad Rizqi Syahal Maulana[1]

[1]Universitas Sugeng Hartono, Sukoharjo, Indonesia

Email: brievaommr22@gmail.com

Abstract. Conventional parking management still relies on manual observation or fragmented dashboards, so field conditions are difficult to monitor quickly and comprehensively. Existing digital parking systems typically display numbers, sensor statuses, or CCTV feeds separately, forcing operators to fuse information mentally; the result is low visibility of slot occupancy, limited monitoring parameters, and no interactive centralized visualization. This paper presents Parkur, a digital twin parking system that unifies the physical parking area, IoT data streams, and an interactive virtual environment in a plug-and-play architecture. The system is organized into eight layers: physical devices, device connectivity and integration, edge intelligence, data platform, digital twin core, analytics and decision, experience dashboard, and a closed feedback loop. Vehicle detection uses the YOLO family of detectors, multi-object tracking uses ByteTrack, and automatic number plate recognition combines YOLO-based plate detection with PaddleOCR, all executed on edge devices such as the NVIDIA Jetson Orin Nano. Slot occupancy is derived from a rule engine with geofencing, and a rule-based alert engine issues real-time notifications. A key innovation is the nearest-slot allocation algorithm, which assigns each entering vehicle the closest available slot to reduce cruising and disorder. The dashboard integrates a 3D visualization area, live CCTV preview, occupancy statistics, and a tap-to-inspect slot inspector. Compared with previous fragmented monitoring approaches, Parkur provides centralized interactive visibility purpose-built for parking operators. The paper details the layered architecture, the edge AI perception pipeline, and the allocation algorithm, and presents the implemented 3D dashboard of the prototype. The main contribution is a complete plug-and-play digital twin architecture for parking management that couples edge AI perception with nearest-slot allocation in one interactive environment.

Received August 2026 / Revised September 2026 / Accepted October 2026

This work is licensed under a Creative Commons Attribution 4.0 International License.

Keywords: Digital Twin; Smart Parking; Automatic Number Plate Recognition; YOLO; Edge AI; Slot Allocation; Internet of Things

INTRODUCTION

Urban parking areas suffer from chronic management problems. Conventional supervision relies on manual observation by attendants or on dashboards that present data in isolation, so the actual field condition is hard to grasp quickly and completely. This leads to inefficiency, security gaps, and misuse of parking areas. Comprehensive reviews confirm that smart parking research has grown rapidly, yet deployment remains fragmented across sensing, guidance, and payment subsystems [1], [2].

Surveys of smart parking solutions classify approaches into sensor-based occupancy detection, guidance, and reservation systems [3], while reviews of guidance, monitoring, and reservation highlight the persistent challenge of integrating these functions [4]. IoT-based prototypes have demonstrated real-time parking systems using ensemble learning [5] and fog computing evaluated through practical case studies [6], proving the technical feasibility of connected parking infrastructure.

The digital twin paradigm offers a unifying abstraction. A digital twin is a virtual replica of a physical asset continuously synchronized through sensor data, a concept whose state of the art spans industry and the built environment [7], with enabling technologies and open challenges mapped systematically [8]. For parking specifically, researchers have combined digital twinning with building information modeling for smart parking management [9], used mobile robots for occupancy detection feeding an urban digital twin [10], proposed digital twin frameworks for urban parking management and mobility forecasting [11], and designed twin-based intelligent parking services for old residential areas [12].

Despite this progress, a gap remains. Most deployed systems still show numbers, sensor states, or CCTV streams separately, leaving operators to combine information themselves. Few works deliver an interactive, centralized visualization purpose-built for parking operations, and none integrate a nearest-position slot allocation strategy that actively guides each entering vehicle. Therefore, this study aims to design Parkur, a digital twin parking system with plug-and-play integration between IoT devices, data flows, and visualization in a single interactive virtual environment. The main objectives are to define a five-layer system architecture, to specify an edge AI perception pipeline for occupancy and plate recognition, to design a nearest-slot allocation algorithm, and to prototype an interactive 3D dashboard for operators.

METHODS

System Architecture

Parkur follows the layered architecture shown in Fig. 1, which comprises eight layers. The physical layer covers parking slots, entrance and exit gates, CCTV cameras, barrier gates, vehicles, indicator lamps, and the operator post. The device connectivity and integration layer provides plug-and-play integration through device adapters, protocol translation, MQTT broker, API gateway, and a device registry. The edge intelligence layer performs local data acquisition and AI inference, including YOLO-based AI detection, ANPR/LPR plate recognition, occupancy logic, and event generation. The data platform layer stores and manages data and events. The digital twin core layer (engine and simulation) maintains entity models, real-time state synchronization, the rule engine, the occupancy engine, the allocation engine, and a simulation engine. The analytics and decision layer provides occupancy analytics, security alerts, and predictions. The experience and dashboard layer delivers the interactive 3D web visualization for operators. A closed feedback loop connects system recommendations and operator actions back to the physical world. Data flow upward from sensors to the dashboard, while control commands flow downward to gates and indicators, forming the bidirectional synchronization that characterizes a digital twin [7].

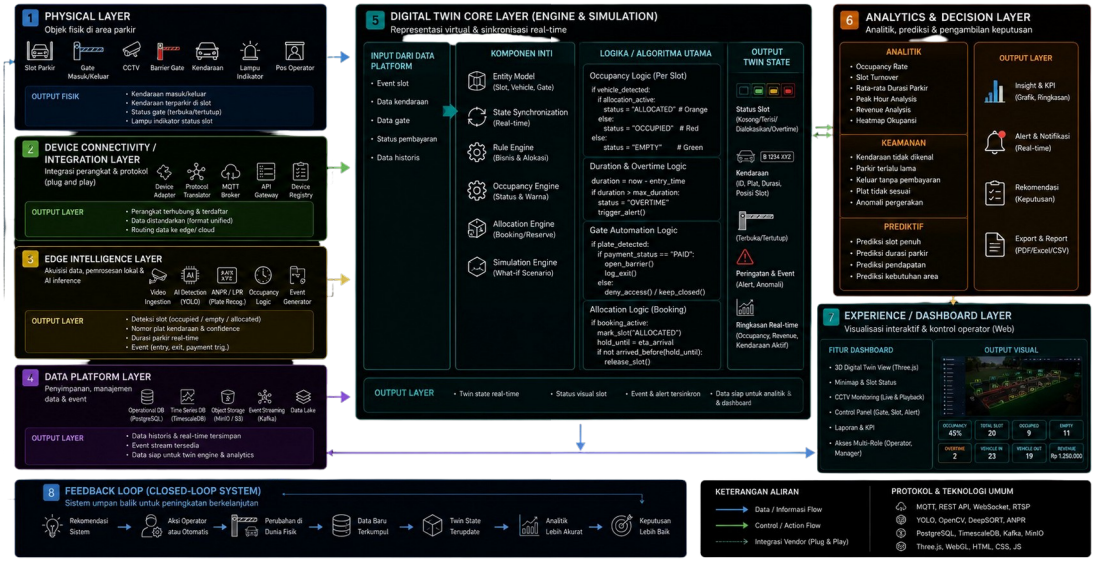


Figure 1. Layered System Architecture of the Parkur Digital Twin Parking System

Physical Layer

Field devices comprise entrance and exit gate cameras (1080p IP cameras, 25 to 30 FPS, RTSP, infrared night vision, field of view of at least 90 degrees) for vehicle detection, plate recognition, and gate triggering; overhead area cameras (1080p to 4MP) for slot occupancy detection; barrier gates with dry-contact, REST, or Modbus control for automatic opening and closing; RGB LED slot indicators with relays showing green, orange, and red status; a mini server or NUC (Intel i5 10th generation or equivalent, 16 GB RAM) hosting the dashboard and middleware; and Gigabit LAN plus Wi-Fi 5 networking. This device mix follows proven real-time IoT parking deployments [6].

IoT Integration Layer

An MQTT-capable device gateway normalizes messages from cameras, gates, and indicators into a common event stream, providing cross-vendor integration without custom drivers per device. New sensors join by publishing to standard topics, which realizes the plug-and-play property of the architecture.

Edge AI Layer

All video inference runs at the edge on an NVIDIA Jetson Orin Nano (8 GB) or, alternatively, an RTX 3050 machine, with 512 GB NVMe storage caching video and events. Processing at the edge follows the vision of edge computing [13] and edge intelligence [14], and mirrors recent real-time surveillance systems that run YOLOv8 on NVIDIA Jetson Nano hardware [15]. The perception pipeline is specified in Table 1.

Table 1. AI Models and Software Stack of Parkur

AI Function	Model	Recommended Version	Input	Output
Vehicle detection	YOLO	YOLOv11n / YOLOv11s	CCTV frame	Vehicle bounding boxes
Slot occupancy	Rule engine + geofence	Internal logic	Bounding boxes + slot coordinates	Free / occupied
Vehicle tracking	ByteTrack	Latest stable	Frame sequence	Vehicle ID
Plate detection	YOLO + OCR	YOLOv11 + PaddleOCR 3.x	Gate frame	Plate region
Plate recognition (ANPR)	PaddleOCR	v3.x	Plate crop	Plate text
Vehicle Re-ID (optional)	FastReID	Latest	Multi-camera	Vehicle identity
Event processing	Node-RED / Python	Python 3.11	Event stream	Triggers
Prediction (optional)	XGBoost	2.x	Parking history	Occupancy forecast
Alert engine	Rule-based	Custom	Real-time status	Notifications

Vehicle detection builds on the YOLO family, from the original unified real-time detector [16] through YOLOv7 [17] to the lightweight YOLOv11 variants selected for edge deployment. Tracking uses ByteTrack, which associates every detection box for robust multi-object tracking [18], succeeding the SORT [19] and DeepSORT [20] trackers. Automatic number plate recognition follows the established two-stage approach of plate detection plus optical character recognition [21], for which YOLO-based detectors have proven robust and real-time capable [22], [23], including layout-independent systems [24] and recent YOLOv8-based designs [25]; edge-optimized ANPR combining YOLOv8 with PaddleOCR has been demonstrated specifically for IoT-based smart parking [26]. Foundational surveys of automatic license plate recognition frame the design choices of this pipeline [21], [24]. The complete perception pipeline, from camera frames to the MQTT event stream, is illustrated in Fig. 2.

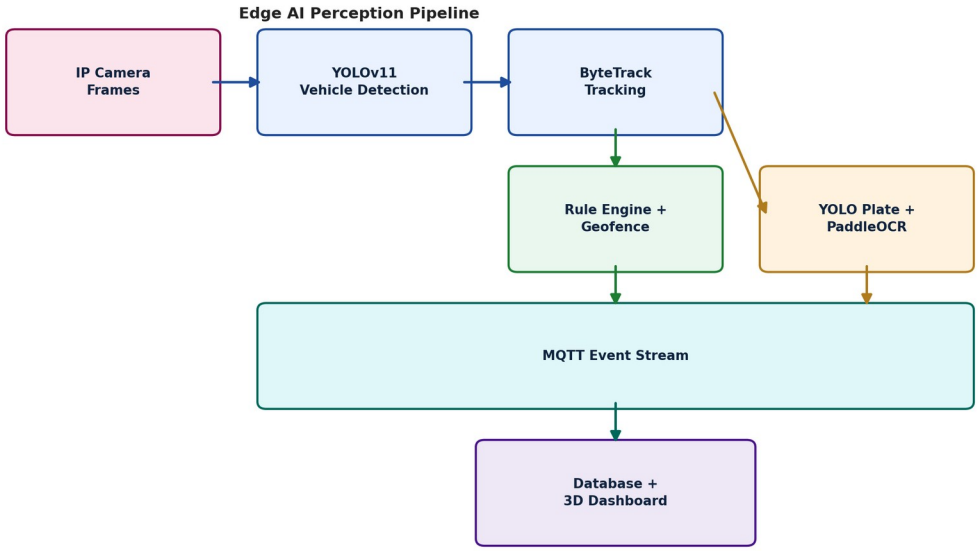


Figure 2. Edge AI Perception Pipeline of Parkur

Nearest-Slot Allocation Algorithm

A key innovation of Parkur is the nearest-slot allocation algorithm. When a vehicle is detected at the entrance gate and its plate is recognized, the system computes the travel distance from the gate to every currently free slot and assigns the slot with the minimum distance. The assignment is pushed to the dashboard, highlighted in the 3D view, and optionally shown on the RGB indicator of the assigned slot. If the slot becomes occupied by another vehicle before arrival, the allocation is recomputed. This active guidance extends the reservation and competition concepts explored in smart parking systems [27] and complements parking guidance research [4] by minimizing cruising distance and keeping parking orderly.

3D Dashboard and Visualization

The dashboard wireframe integrates a 3D visualization area rendering the parking lot as an interactive digital twin, a live CCTV preview panel, occupancy statistics, and a slot onTap inspector that reveals details of any selected slot. Entrance and exit gates and individual parking slots are represented as interactive objects with primary and additional parameters. This centralized interactive visualization directly addresses the fragmentation problem identified in the literature [1], [2], following the twin-based visualization direction of recent parking research [9], [11].

Optional Extensions

Two optional modules are specified for future activation. Vehicle re-identification with FastReID links identities across multiple cameras, building on deep-learning re-identification surveys [28], [29]. Occupancy prediction with XGBoost forecasts demand from historical data, following evolutionary deep learning approaches to car park occupancy prediction [30] and XGBoost-LSTM model fusion demonstrated for truck parking [31].

RESULT AND DISCUSSION

System Specification Summary

The minimum component specifications derived from the design are summarized in Table 2. The specification balances real-time inference requirements against deployment cost by offering the Jetson Orin Nano as the primary edge device and an RTX 3050 machine as an alternative.

Table 2. Minimum Component Specifications of Parkur

Layer	Component	Minimum Specification	Function
Physical	Gate cameras	1080p, 25–30 FPS, RTSP,	Vehicle detection, ANPR,

		IR	gate trigger
Physical	Area cameras	1080p–4MP, overhead angle	Slot occupancy detection
Physical	Barrier gate	Dry contact / REST / Modbus	Automatic open–close control
Physical	Slot indicators	RGB LED + relay	Green–orange–red status
Physical	Mini server / NUC	i5 10th gen+, 16 GB RAM	Dashboard and middleware host
Edge	Edge AI device	Jetson Orin Nano 8 GB or RTX 3050	Real-time AI inference
Data	Database server	4 vCPU, 16 GB RAM, 500 GB SSD	State and history storage
Dashboard	Client display	Full HD 1920×1080	Operator monitoring

Prototype Implementation

The Parkur dashboard prototype was implemented as shown in Fig. 3. The 3D visualization renders the parking area at night with labeled slots A1–A10 and B1–B10, occupied slots highlighted in red and free slots in green. The left panel displays the current time, night mode toggle, total slots (20), occupied slots (8), empty slots (12), occupancy rate (40%), a live gate-camera CCTV preview, and traffic statistics. The top-right minimap gives an overview of the whole area, and the slot onTap inspector reveals per-slot details. The prototype confirms that occupancy state, video evidence, and per-slot details coexist in one interactive operator view.



Figure 3. Implemented 3D Dashboard of the Parkur Prototype (Night Mode)

Discussion

Compared with conventional systems that display numbers, sensor states, or CCTV separately [1], [2], Parkur fuses all sources into one interactive twin, eliminating the mental integration burden on operators. Compared with existing digital twin parking proposals focused on modeling [9] or forecasting [11], Parkur adds an operational loop: edge AI perception, nearest-slot allocation, and actuation of gates and

indicators. The edge-based ANPR design is consistent with recent demonstrations of YOLO-based plate recognition on Jetson-class hardware [26], [15]. The plug-and-play MQTT layer lowers integration cost when extending the system to new areas or camera vendors. The main limitation is that the system is currently at the design and wireframe prototype stage; quantitative evaluation of detection accuracy, end-to-end latency, and allocation efficiency in a live parking lot remains future work, alongside activation of the re-identification and prediction modules and a longitudinal operational study.

CONCLUSION

This paper presented Parkur, a digital twin parking system that unifies IoT sensing, edge AI perception, and interactive 3D visualization in a five-layer plug-and-play architecture. The system specifies YOLO-based vehicle detection, ByteTrack tracking, and PaddleOCR plate recognition on edge devices, derives slot occupancy through a rule engine with geofencing, and introduces a nearest-slot allocation algorithm that assigns each entering vehicle the closest free slot. The dashboard prototype integrates a 3D visualization area, live CCTV preview, occupancy statistics, and a slot inspector in a single operator view. By replacing fragmented dashboards with one synchronized virtual environment, Parkur offers a practical path toward more orderly, visible, and secure parking operations, with future work directed at live deployment and quantitative evaluation. The edge-centric design follows the same direction as digital-twin-driven optimization frameworks that place computation at the edge of Industrial IoT architectures [32], and edge-based real-time anomaly detection shows that lightweight models at the edge can meet strict latency budgets [33].

REFERENCES

[1] A. Fahim, M. Hasan, and M. A. Chowdhury, "Smart parking systems: Comprehensive review based on various aspects," Heliyon, vol. 7, no. 5, e07050, May 2021.

[2] C. Biyik, Z. Allam, G. Pieri, D. Moroni, M. O'Fraifer, E. O'Connell, S. Olariu, and M. Khalid, "Smart parking systems: Reviewing the literature, architecture and ways forward," Smart Cities, vol. 4, no. 2, pp. 623–642, 2021.

[3] T. Lin, H. Rivano, and F. Le Mouël, "A survey of smart parking solutions," IEEE Trans. Intell. Transp. Syst., vol. 18, no. 12, pp. 3229–3253, Dec. 2017.

[4] A. O. Kotb, Y. Shen, and Y. Huang, "Smart parking guidance, monitoring and reservations: A review," IEEE Intell. Transp. Syst. Mag., vol. 9, pp. 6–16, 2017.

[5] W. H. Elashmawi, A. Akram, M. Yasser, M. Hisham, M. Mohammed, N. Ihab, A. Ali, et al., "IOT based smart parking system using ensemble learning," Intell. Autom. Soft Comput., vol. 36, no. 3, pp. 3637–3656, 2023.

[6] M. A. Ala'anzy, A. Abilakim, R. Zhanuzak, L. Li, et al., "Real time smart parking system based on IoT and fog computing evaluated through a practical case study," Sci. Rep., vol. 15, p. 33483, 2025.

[7] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, "Digital twin in industry: State-of-the-art," IEEE Trans. Ind. Informat., vol. 15, no. 4, pp. 2405–2415, Apr. 2019.

[8] A. Fuller, Z. Fan, C. Day, and C. Barlow, "Digital twin: Enabling technologies, challenges and open research," IEEE Access, vol. 8, pp. 108952–108971, 2020.

[9] J. K. Coching, R. K. C. Billones, A. K. M. Brillantes, S. Yunus, V. A. Pitogo, and R. Senkerik, "Digital twinning mechanism and building information modeling for a smart parking management system," Smart Cities, vol. 8, no. 5, p. 146, 2025.

[10] L. Kühn, S. Sir, B. Schäfer, and H. Tchouankem, "Parking space occupancy detection using a mobile robot for an urban digital twin," in Proc. 2025 IEEE Eur. Technol. Eng. Manage. Summit (E-TEMS), pp. 258–263, 2025.

[11] F. Piccialli et al., "A digital twin framework for urban parking management and mobility forecasting," Nature Commun., vol. 16, no. 1, p. 9400, Oct. 2025.

[12] W. Chen, X. Wang, and M. Wu, "Intelligent parking service system design based on digital twin for old residential areas," Electronics, vol. 13, no. 23, p. 4597, 2024.

[13] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," IEEE Internet Things J., vol. 3, no. 5, pp. 637–646, Oct. 2016.

[14] D. Xu, T. Li, Y. Li, X. Su, S. Tarkoma, T. Jiang, J. Crowcroft, and P. Hui, "Edge intelligence: Empowering intelligence to the edge of network," Proc. IEEE, vol. 109, no. 11, pp. 1778–1837, Nov. 2021.

[15] V. Ghodmare, T. Kakade, S. Khan, S. Aghor, and P. Rane, "Edge computing-based real-time surveillance system with YOLOv8 object detection using NVIDIA Jetson Nano," in Proc. 2025 5th Int. Conf. Pervasive Comput. Social Netw. (ICPCSN), 2025.

[16] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, NV, USA, 2016, pp. 779–788.

[17] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Vancouver, BC, Canada, 2023, pp. 7464–7475.

[18] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang, "ByteTrack: Multi-object tracking by associating every detection box," in Proc. Eur. Conf. Comput. Vis. (ECCV), Tel Aviv, Israel, 2022.

[19] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in Proc. IEEE Int. Conf. Image Process. (ICIP), Phoenix, AZ, USA, 2016, pp. 3464–3468.

[20] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in Proc. IEEE Int. Conf. Image Process. (ICIP), Beijing, China, 2017, pp. 3645–3649.

[21] S. Du, M. Ibrahim, M. Shehata, and W. Badawy, "Automatic license plate recognition (ALPR): A state-of-the-art review," IEEE Trans. Circuits Syst. Video Technol., vol. 23, no. 2, pp. 311–325, Feb. 2013.

[22] R. Laroca, E. Severo, L. A. Zanlorensi, L. S. Oliveira, G. R. Gonçalves, W. R. Schwartz, and D. Menotti, "A robust real-time automatic license plate recognition based on the YOLO detector," in Proc. Int. Joint Conf. Neural Netw. (IJCNN), Rio de Janeiro, Brazil, 2018, pp. 1–10.

[23] R. Laroca, L. A. Zanlorensi, G. R. Gonçalves, E. Todt, W. R. Schwartz, and D. Menotti, "An efficient and layout-independent automatic license plate recognition system based on the YOLO detector," IET Intell. Transp. Syst., vol. 15, no. 4, pp. 483–503, 2021.

[24] J. Shashirangana, H. Padmasiri, D. Meedeniya, and C. Perera, "Automated license plate recognition: A survey on methods and techniques," IEEE Access, vol. 9, pp. 11203–11225, 2020.

[25] A. Amin, R. Mumtaz, M. J. Bashir, and S. M. H. Zaidi, "Next-generation license plate detection and recognition system using YOLOv8," in Proc. 2023 IEEE 20th Int. Conf. Smart Communities: Improving Quality of Life using AI, Robot. IoT (HONET), 2023.

[26] M. Ebrahimishadman and A. Souri, "Edge-optimized automatic number plate recognition for IoT-based smart parking using YOLOv8," Comput. Mater. Contin., 2026.

[27] Z. Hu, Y. Cao, J. Huang, L. Wu, L. Xu, and R. Zhang, "CRPark: A smart parking system with competition and reservation services," IEEE Trans. Veh. Technol., vol. 74, no. 8, pp. 12981–12996, 2025.

[28] H. Wang, J. Hou, and N. Chen, "A survey of vehicle re-identification based on deep learning," IEEE Access, vol. 7, pp. 172443–172469, 2019.

[29] S. D. Khan and H. Ullah, "A survey of advances in vision-based vehicle re-identification," Comput. Vis. Image Underst., vol. 182, pp. 50–63, 2019.

[30] A. Camero, J. Toutouh, D. H. Stolfi, and E. Alba, "Evolutionary deep learning for car park occupancy prediction in smart cities," in Proc. Int. Conf. Learn. Intell. Optim. (LION), Kalamata, Greece, 2018, pp. 386–401. Springer, 2018.

[31] S. Gutmann, C. Maget, M. Spangler, and K. Bogenberger, "Truck parking occupancy prediction: XGBoost-LSTM model fusion," Frontiers Future Transp., vol. 2, p. 693708, 2021.

[32] S. Suyahman, D. U. Iswavigra, H. Wibowo, A. B. Trisnawan, A. Wicaksono, et al., "Digital Twin Driven Real Time Performance Optimization of Smart Factory Production Systems Using Edge Computing and Industrial Internet of Things Architecture," Int. J. Industrial Innovation and Mechanical Engineering, vol. 1, no. 2, 2024.

[33] D. Prasetyo, S. Suyahman, R. Y. Widiastuti, M. Mursalim, and A. Pribadi, "Edge Computing Enabled Real Time Anomaly Detection Framework for Secure Industrial Cyber Physical Systems Using Lightweight Deep Neural Networks," Int. J. Mechanical, Industrial and Control Systems Engineering, 2024.