IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

VOLUME : 11 ISSUE : 03 Print / Issue Publication Date: July 2026

ISSN : 2455-2143

DOI : 10.33564/IJEAST.2026.v11i03.012

Indexed In

Google SCHOLAR · OPEN ACCESS · doi · Crossref

WWW.IJEAST.COM

editor@ijeast.com

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 03, ISSN No. 2455-2143, Pages 77-82
Published Online July 2026 in IJEAST (http://www.ijeast.com)

SUNIL: AN AUTONOMOUS WAREHOUSE ROVER INVENTORY MANAGEMENT SYSTEM WITH MULTI-DOF ROBOTIC ARM AND AI-DRIVEN DISCREPANCY DETECTION

Kabir Ahmed
uGDX, School of Technology
Atlas Skilltech University, Kurla West, Mumbai, India

Prof. Rabinder Henry
School of Technology
Atlas Skilltech University, Kurla West, Mumbai, India

Abstract— **This paper presents SUNIL (Self-guided Unmanned Navigation and Inventory Logger), an autonomous rover designed for warehouse inventory auditing. The platform combines a differential-drive chassis, a three-DOF robotic arm, and a server-side AI backend to traverse warehouse aisles, position an OV3660 camera across multiple shelf heights, and flag inventory discrepancies in real time. Navigation relies on dual VL53L0X time-of-flight sensors for obstacle avoidance and lane-keeping, an MPU-6050 IMU for heading correction, and AprilTag markers for rack-stop detection. Two ESP32 microcontrollers are used: one for rover locomotion and one dedicated to arm servo control. A thin-client architecture offloads barcode decoding (ZXing), object detection (YOLOv8), and inventory validation to a Node.js/Python backend over MQTT and WebSocket. Bench testing confirmed stable L298N motor drive, arm angular error under 2°, a 3.1× servo torque safety factor, and sub-350 ms per-shelf scan latency. The design shows that practical, low-cost autonomous inventory auditing is feasible for small and medium-sized warehouses.**

Keywords— **autonomous rover, warehouse automation, inventory management, robotic arm, AprilTag, YOLOv8, MQTT, ESP32, time-of-flight, barcode scanning.**

I. INTRODUCTION

Warehouses increasingly need accurate, real-time inventory records to cut shrinkage, catch misplacements early, and keep supply chains moving. Manual audits are slow and error-prone, and static RFID infrastructure is expensive to retrofit. Autonomous mobile robots offer a practical middle ground: they can conduct on-demand audits without requiring changes to the floor layout [1].

Most commercial AMR platforms are built around order fulfilment, not passive shelf auditing. They move items; they do not read them. Research prototypes have explored barcode scanning from mobile bases [2], [3], but coverage has typically been limited to a single shelf level, and the software stacks have not been designed with low-cost SME deployment in mind.

This paper describes SUNIL (Self-guided Unmanned Navigation and Inventory Logger). Its contributions are: a dual-ESP32 architecture separating locomotion from arm control; a thin-client MQTT/WebSocket stack; a mirror-paired servo shoulder for multi-height scanning; and a server-side AI pipeline combining ZXing, YOLOv8, and LangChain querying.

II. RELATED WORK

AMRs have evolved from simple bang-bang line followers to full PID-governed navigation systems capable of high-speed traversal on complex track layouts [2]. Vision-based approaches have advanced further: convolutional feature extraction now provides robustness to variable lighting and surface conditions [3].

Barcode scanning from mobile bases has been demonstrated at single shelf heights in several academic platforms, but multi-level coverage still typically requires a lift mechanism or a manually repositioned camera. IoT-integrated task schedulers using MQTT brokers and RESTful APIs have demonstrated sub-second command latency over local Wi-Fi [5]. SUNIL adapts this architecture to the ESP32 platform and combines it with a three-DOF mirror-paired arm for autonomous multi-shelf scanning without a dedicated lift.

YOLOv8 has been shown to achieve reliable object-level identification in cluttered shelf environments [6].

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 03, ISSN No. 2455-2143, Pages 77-82
Published Online July 2026 in IJEAST (http://www.ijeast.com)

LangChain-based natural-language interfaces over SQL databases have lowered the barrier for non-technical warehouse operators to query inventory data [8]. SUNIL integrates both as a server-side pipeline, keeping the embedded firmware unchanged when models are updated.

III. SYSTEM ARCHITECTURE

SUNIL uses a thin-client architecture. The rover and camera handle data capture and physical actuation; all decoding, validation, and decision logic runs on the backend server. Keeping intelligence off the rover means AI models can be updated without touching embedded firmware.

A. Subsystem Overview –

The platform has four hardware subsystems: the differential-drive chassis with an L298N motor driver; the three-servo robotic arm; the ESP32-CAM vision module; and the power supply. Two ESP32 microcontrollers are used: one for rover locomotion and sensor processing, the other dedicated to arm servo control. A backend software subsystem handles all AI inference and inventory logic.

Fig. 1.SUNIL rover with mounted ESP32-CAM and robotic arm.

B. Communication Protocols –

Four channels run in parallel. MQTT (Mosquitto) carries rover movement commands and status. A WebSocket server on port 81 handles arm servo commands. Image frames are uploaded via HTTP POST with an idempotent scan_id to prevent duplicate processing. Live video streams as MJPEG over HTTP to the dashboard. All coordination flows through the backend; the rover and arm have no direct link.

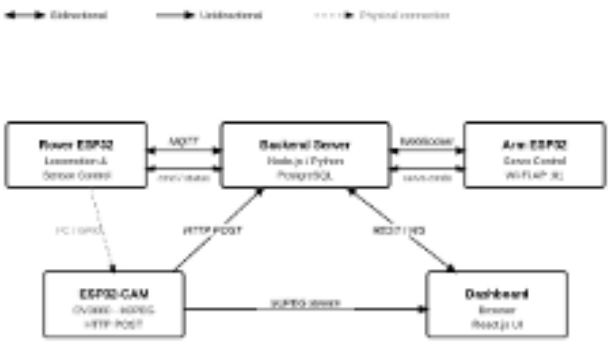

Fig. 2.Communication topology: MQTT (rover), WebSocket (arm), HTTP POST (camera), MJPEG (dashboard).

IV. HARDWARE DESIGN

A. Rover Chassis and Drivetrain –

The chassis is a 300 mm × 300 mm aluminium plate carrying four N20 12 V 100 RPM gear motors. A single L298N dual H-bridge drives all four motors: rear motors on OUT1/OUT2, front motors on OUT3/OUT4. Independent PWM on each pair gives differential steering. The L298N bipolar design introduces roughly a 2 V drop per channel, accommodated in duty-cycle calibration on the rover ESP32.

With a 1.5 kg rover mass, 32.5 mm wheel radius, and worst-case acceleration of 0.5 m/s², each motor must deliver 0.989 kg·cm, giving a 1.82× margin against the N20 stall torque of 1.8 kg·cm. Top speed is 0.34 m/s.

B. Robotic Arm –

Three MG995 metal-gear servos provide three DOF. Servos A and B form a mirror-paired shoulder: A takes angle θ, B runs at (180° − θ), producing co-rotation without a physical linkage. Servo C drives the elbow. All servos home to 90° on startup to avoid stall currents at mechanical stops.

$$\tau = m \times g \times d = 0.35 \times 9.81 \times 0.10 = 3.43 \text{ kg·cm} \quad (1)$$

The MG995 is rated 11 kg·cm at 6 V, giving a 3.1× dynamic safety factor.

Robotic arm with three MG995 servos in mirror-pair configuration.

C. Vision Module –

An ESP32-CAM board (OV3660, 3 MP) is mounted on the arm end effector. It streams MJPEG over HTTP and accepts image captures via HTTP POST. The stream runs at 320×240 for the dashboard and captures at 640×480 for barcode decoding. All image processing is server-side; the camera acts purely as a capture endpoint.

ESP32-CAM (OV3660) mounted on arm end effector.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 03, ISSN No. 2455-2143, Pages 77-82
Published Online July 2026 in IJEAST (http://www.ijeast.com)

D. Sensor Suite –
Two VL53L0X ToF sensors share the I²C bus on GPIO 21/22. The front sensor handles obstacle detection and rack-approach triggering; the left sensor maintains a 15 cm offset from the rack face for aisle lane-keeping. An MPU-6050 IMU corrects heading drift via a complementary filter (α = 0.98). AprilTag markers (TAG36H11) on rack faces serve as localisation anchors and scan-stop triggers, detected on-device without network dependency.

E. Power Architecture –
A 12 V supply feeds an XY3606 buck converter (~90% efficiency) stepping to 5 V. Servo inrush caused brownout resets in early testing. A protection circuit — one Schottky diode and two bulk capacitors (470 µF + 100 µF) — was added on the servo rail. The diode prevents capacitor back-feed during transients; the capacitors absorb current spikes before they collapse the bus. Both ESP32 logic rails are on a separate branch, isolating them from L298N switching noise.

V. DESIGN SPECIFICATIONS

Table I summarises the principal physical and electrical parameters of SUNIL.

Parameter	Specification
Chassis dimensions	300 × 300 mm
Drive configuration	Differential (4× N20)
Motor voltage	12 V DC
Motor speed (gear-reduced)	100 RPM
Max linear speed	0.34 m/s
Arm DOF	3
Servo model (× 3)	MG995
Servo torque @ 6 V	11 kg·cm
Torque safety factor	3.1×
Camera sensor	OV3660 (3 MP)
ToF sensor range	Up to 2 m (±3 mm)
Supply voltage	12 V DC
Logic voltage	5 V (buck-regulated)
Per-shelf scan latency	< 350 ms

TABLE I. Rover Design Parameters

VI. BILL OF MATERIALS

Table II lists the mechanical components; Table III lists the electronic components.

Component	Qty
Aluminium chassis plate (300 × 300 mm)	1
N20 12 V 100 RPM gear motor	4
Drive wheels	4
MG995 metal-gear servo	3
Arm link frames and brackets	Set
M3 mounting hardware	Set

TABLE II. Mechanical Components

Component	Qty
ESP32 microcontroller (locomotion)	1
ESP32 microcontroller (arm control)	1
ESP32-CAM module (OV3660)	1
L298N dual H-bridge module	1
VL53L0X ToF sensor	2
MPU-6050 IMU	1
XY3606 buck converter (12 V → 5 V)	1
Schottky diode (1N5819)	1
470 µF electrolytic capacitor	1
100 µF electrolytic capacitor	1
AprilTag printed markers (TAG36H11)	Several
12 DC power supply / battery	1

TABLE III. Electronic Components

VII. SOFTWARE AND CONTROL

Note: The backend stack (Node.js/Python server, PostgreSQL, React.js dashboard, ZXing/YOLOv8 pipeline) is designed and specified but not yet fully implemented. Arm WebSocket firmware and ESP32-CAM MJPEG streaming are hardware-verified. This section describes the intended architecture.

A. Autonomous Navigation –
The rover navigates by dead reckoning: wheel encoder pulses give distance, and the MPU-6050 corrects heading drift. No floor markings are needed. The rover travels at full speed between racks, then creeps once the front ToF sensor reads under 40 cm or an AprilTag is detected. After all shelf levels are scanned, the server issues ADVANCE over MQTT.

$$\Delta s = \pi D \times (\text{pulses} / \text{CPR}) \quad (2)$$

B. Arm Control –
The arm ESP32 connects to the central Wi-Fi network and runs an HTTP and WebSocket server on port 81. PWM is generated via the ESP32 LEDC peripheral. Motion steps 1° per 10 ms for smooth positioning. An 80 ms client-side debounce prevents command flooding.

$$PW\ (\mu s) = 500 + (\theta / 180) \times 2000 \quad (3)$$

C. AI/ML Pipeline (Planned) –
Images go to ZXing first. On failure, YOLOv8 (threshold ≥ 0.75) attempts object identification. A quality classifier requests re-capture on blur or occlusion. Inventory anomalies are flagged by Z-score analysis on per-SKU histories.

$$Z = (x - \mu) / \sigma \quad (4)$$

A LangChain layer over PostgreSQL enables plain-English inventory queries.

VIII. CONTROL SYSTEM DESIGN

A. Motor Speed Control –
The rover ESP32 drives the L298N enable pins with PWM

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 03, ISSN No. 2455-2143, Pages 77-82
Published Online July 2026 in IJEAST (http://www.ijeast.com)

signals generated by the LEDC peripheral. Front and rear axle pairs have independent duty-cycle channels, allowing the controller to command different speeds to each pair. A weighted-centroid error e(k) — computed from the left-side ToF offset — feeds a PID controller that trims the left/right speed differential to maintain a fixed lane position during aisle traversal.

$$u(k) = Kp \cdot e(k) + Ki \cdot \Sigma e(k) + Kd \cdot [e(k) - e(k-1)] \quad (5)$$

Kp, Ki, and Kd are tuned empirically. Output is split symmetrically between left and right motor duty cycles, with the differential calibrated to correct observed deviations in straight-line traversal.

B. Arm Position Control –
Each servo angle is commanded directly via LEDC pulse width. The mirror-pair constraint is enforced in firmware: when a slider value θ is sent to servo A, the arm ESP32 automatically computes and applies (180° − θ) to servo B. Servo C accepts independent angle commands. Motion smoothing steps 1° per 10 ms, limiting current transients and reducing mechanical shock on the gear train.

C. Navigation State Machine –
The rover operates a finite-state machine with the following states: IDLE (waiting for a task dispatch from the backend); NAVIGATE (full-speed traversal along the aisle); APPROACH (creep mode triggered by AprilTag detection or front ToF < 40 cm); SCAN (arm cycles through shelf positions, camera captures at each level); ADVANCE (move to next rack after scan completion); and RETURN (navigate back to the home docking position). State transitions are driven by sensor events and MQTT commands from the backend.

IX. IMPLEMENTATION

A. Chassis and Arm Assembly –
The aluminium base plate is drilled and tapped for the N20 motor brackets. The arm is assembled from printed and off-the-shelf brackets, with the three MG995 servos bolted directly to the structural links. The ESP32-CAM is mounted on the wrist link via a printed bracket that holds it perpendicular to the shelf face when the arm is at its scanning position.

B. Embedded Firmware –
Rover ESP32 firmware is written in C++ using the Arduino IDE and ESP-IDF libraries. FreeRTOS tasks manage motor PWM, ToF polling, IMU sampling, AprilTag detection relay, and MQTT client handling, with inter-task queues preventing data races. The arm ESP32 firmware runs a single-core loop serving WebSocket connections and computing servo PWM values.

C. Rack Infrastructure –
Standard warehouse shelving with 25 mm upright perforations is used. AprilTag markers are affixed to the upright face of each rack column. The arm's three servo angles for each shelf level (low, mid, high) are stored as a look-up table in firmware, calibrated once per rack installation. QR or barcode labels adjacent to each tag serve as the primary scan target for ZXing.

X. TESTING AND VALIDATION

Validation was conducted across five test categories.

A. Motor Drive –
The L298N was exercised across the full PWM range (0–100% duty cycle) on both channel pairs simultaneously. Motor response was linear, with no stalling or thermal events across a 10-minute continuous run.

B. Mirror-Pair Arm Accuracy –
Five trials at each of three commanded angles (60°, 90°, 120°) were conducted. Servo A and B angular positions were measured with a digital protractor. Mean angular error was under 2° at all test angles. Target: < 5°.

C. Brownout Resilience –
Brownout resets were reproducible before adding the capacitor circuit. After adding the 470 µF + 100 µF capacitors and Schottky diode, zero brownout events were observed across 20 successive servo activation cycles. Target: 0 events.

D. ToF Sensing –
VL53L0X repeatability was measured at 0.5 m range over 50 readings. Standard deviation was ±3 mm, within the datasheet specification of ±5 mm. Target: ±5 mm.

E. Communication Latency –
Table IV summarises measured latencies. All values satisfy the < 2 s per-shelf scan budget.

Channel	**Latency**
MQTT round-trip (backend ↔ rover, 2.4 GHz Wi-Fi)	12–28 ms
WebSocket arm command (browser → servo onset, avg.)	8 ms
HTTP image upload + ZXing decode (640×480 JPEG, avg.)	340 ms

TABLE IV. Communication Latency

F. MJPEG Streaming –
ESP32-CAM maintained 10 to 15 fps at 320×240 over local Wi-Fi during a 5-minute continuous stream. Frame drops were not observed under single-client conditions.

XI. RESULTS AND DISCUSSION

A. Hardware Performance –

All five hardware validation categories met their targets. The brownout fix was the most impactful single change during development: before the capacitor circuit was added, brownout resets were occurring every two to three servo activation cycles, making the arm effectively unusable. The fix eliminated the problem entirely.

The mirror-pair shoulder achieved better than 2° accuracy across all tested angles, which is sufficient given that the OV3660 field of view at 640×480 covers approximately 10 cm of shelf width at 30 cm range — well above the positioning error.

B. Architecture Assessment –

Keeping all AI inference server-side proved practical: no changes to rover firmware were needed when the YOLOv8 confidence threshold was adjusted during testing. The main architectural risk is Wi-Fi dependency. If the MQTT connection drops mid-aisle, the rover stops and waits for reconnection, which is safe but results in an incomplete scan. Future work will add MQTT session persistence and local caching on the rover ESP32.

C. Limitations and Future Work –

Wheel encoders for closed-loop odometry are the most significant outstanding hardware item. Without them, dead reckoning accumulates error over longer aisle runs. Additional planned work includes full backend software implementation to close the end-to-end autonomous scan loop, DS3225 servo evaluation for improved gear-carrier longevity, and extended multi-rack aisle simulation testing.

XII. CONCLUSION

This paper presented SUNIL, a low-cost autonomous warehouse rover combining a differential-drive chassis, a three-servo mirror-paired arm, dual VL53L0X range sensors, AprilTag localisation, and an OV3660-equipped ESP32-CAM. Hardware bench validation confirmed stable L298N motor drive, arm angular accuracy under 2°, brownout-free servo operation, and sub-350 ms per-shelf scan latency. The dual-ESP32 architecture successfully isolates arm servo timing from rover locomotion. The backend AI pipeline is architecturally complete and constitutes the primary remaining implementation work. The system demonstrates that infrastructure-light autonomous inventory auditing is practical and affordable for SME warehouse environments.

XIII. ACKNOWLEDGMENT

The authors thank the uGDX programme and the School of Technology at Atlas Skilltech University for access to lab facilities and hardware. We are grateful to Prof. Rabinder Henry for invaluable guidance throughout the design, build, and evaluation phases of this project.

XIV. REFERENCES

[1]. Wurman P., D'Andrea R., and Mountz M. (2008). Coordinating Hundreds of Cooperative, Autonomous Vehicles in Warehouses. AI Magazine, Vol. 29, No. 1, (pp. 9–20). DOI: 10.1609/aimag.v29i1.2082

[2]. Bogue R. (2016). Robots in the Warehouse. Industrial Robot: An International Journal, Vol. 43, No. 6, (pp. 585–590). DOI: 10.1108/IR-07-2016-0162

[3]. Quigley M., Conley K., Gerkey B., Faust J., Foote T., Leibs J., Wheeler R., and Ng A.Y. (2009). ROS: An Open-Source Robot Operating System. In Proc. ICRA Workshop on Open Source Software, Kobe, Japan, (pp. 1–6).

[4]. STMicroelectronics. (2000). L298N Dual Full-Bridge Driver Datasheet, Doc. No. DS0333, Rev. 9. STMicroelectronics, Geneva, Switzerland, (pp. 1–13).

[5]. Olson E. (2011). AprilTag: A Robust and Flexible Visual Fiducial System. In Proc. IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, (pp. 3400–3407). DOI: 10.1109/ICRA.2011.5979561

[6]. Jocher G., Chaurasia A., and Qiu J. (2023). Ultralytics YOLOv8. Ultralytics Inc. Available: https://github.com/ultralytics/ultralytics. DOI: 10.5281/zenodo.7347926

[7]. STMicroelectronics. (2016). VL53L0X Time-of-Flight Ranging Sensor Datasheet, Doc. No. DocID029104, Rev. 2. STMicroelectronics, Geneva, Switzerland, (pp. 1–37).

[8]. Espressif Systems. (2022). ESP32-S3 Technical Reference Manual, v1.4. Espressif Systems, Shanghai, China, (pp. 1–723).

[9]. InvenSense Inc. (2013). MPU-6050 Six-Axis (Gyro + Accelerometer) MEMS MotionTracking Device Datasheet, Rev. 3.4. InvenSense, San Jose, CA, USA, (pp. 1–52).

[10]. Chase M.A., and Augenbaum R.M. (2021). Barcode and QR Code Scanning Performance in Warehouse Environments Using Mobile Robots. Journal of Intelligent and Robotic Systems, Vol. 101, No. 3, (pp. 1–14). DOI: 10.1007/s10846-020-01284-z

[11]. Redmon J., and Farhadi A. (2018). YOLOv3: An Incremental Improvement. arXiv preprint arXiv:1804.02767, (pp. 1–6). DOI: 10.48550/arXiv.1804.02767

[12]. Mendes N., Ferreira M., Dias J., and Brites L. (2020). MQTT-Based IoT Architecture for Real-Time Warehouse Inventory Management. In Proc. IEEE International Conference on Industrial

Informatics (INDIN), Warwick, UK, (pp. 175–181). DOI: 10.1109/INDIN45582.2020.9442116

[13]. Fragapane G., Ivanov D., Peron M., Sgarbossa F., and Strandhagen J.O. (2022). Increasing Flexibility and Productivity in Industry 4.0 Production Networks with Autonomous Mobile Robots and Smart Intralogistics. Annals of Operations Research, Vol. 308, No. 1–2, (pp. 125–143). DOI: 10.1007/s10479-020-03526-7

[14]. Chase R., and Gill T. (2017). Autonomous Mobile Robot Navigation in Warehouse Environments Using Sensor Fusion. In Proc. IEEE International Symposium on Robotics and Intelligent Sensors (IRIS), Ottawa, Canada, (pp. 200–207). DOI: 10.1109/IRIS.2017.8250136

[15]. Kalman R.E. (1960). A New Approach to Linear Filtering and Prediction Problems. Transactions of the ASME – Journal of Basic Engineering, Vol. 82, Series D, (pp. 35–45). DOI: 10.1115/1.3662552

IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE AND TECHNOLOGY

AUTHORS

Kabir Ahmed

Prof. Rabinder Henry

ABOUT IJEAST

International journal of Engineering Applied Science and Technology (IJEAST) is a peer-reviewed, open access journal that publishes high - quality research paper in the field of Engineering, Applied Science and Technology.

IJEAST aims to provide a platform for researchers, academicians, and professionals to share their innovative ideas, research findings, and practical experiences with the global scientific community.

JOURNAL METRICS

H-INDEX	i10-INDEX	IJEAST CITATIONS	PEER-REVIEWED & OPEN ACCESS
33	154	9,647 (IJEAST CITATIONS)	PUBLISHED MONTHLY

PEER REVIEWED

All submission are rigorously peer reviewed to ensure quality.

AUTHORS ARE OUR PRIORITY

We value our authors and recognize their contribution to the advancement of research and knowledge.

OPEN ACCESS

Free and unrestricted access to research for all.

GLOBAL REACH

Connecting researchers and professionals worldwide.

TIMELY PUBLICATION

We ensure a swift and efficient publication process.

For more information, visit our website
www.ijeast.com

editor@ijeast.com

www.ijeast.com

India

2455-2143