iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** VI **Month of publication:** June 2026

DOI: https://doi.org/10.22214/ijraset.2026.83466

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VI June 2026- Available at www.ijraset.com

Intelligent Parcel Sorting System Using DIGIPIN-Based Edge AI and OCR

Hemalatha[1], Srinikesh[2], Prajit[3], Venkat Srinath[4], Surendran[5]
Department of MechatronicsPSG Polytechnic College, Coimbatore, Tamil Nadu, India

Abstract—The rapid expansion of e-commerce has created unprecedented demand for high-throughput, error-free parcel sorting in logistics hubs. Conventional sorting systems that rely on barcodes or QR codes introduce a redundant encoding step and fail when labels are damaged or misaligned. This paper presents a fully automated, cloud-independent parcel sorting conveyor system that leverages Edge Artificial Intelligence and pure Optical Character Recognition (OCR) to read a 10-character alphanumeric DIGIPIN (Digital Postal Index Number) directly from package surfaces. The system employs a dual-node distributed architecture: a Raspberry Pi 5 acting as the cognitive Edge AI node for image processing, OCR inference, and GUI rendering, and an Arduino UNO serving as the deterministic real-time execution node for sensor polling, motor control, and servo actuation. Node communication is achieved wirelessly via an HC-05 Bluetooth serial link. Prototype testing demonstrated 100% sorting accuracy on clean labels and 86.7% accuracy across real-world varied label conditions, achieving a cycle time of approximately 4 seconds (≈900 parcels/hour). The proposed system provides a scalable, low-cost alternative to cloud-dependent industrial sorters suitable for India's expanding postal and logistics networks.

Keywords—DIGIPIN, Edge AI, OCR, Parcel Sorting, Raspberry Pi, Arduino, Conveyor Automation, IoT.

I. INTRODUCTION

The landscape of global supply chain management has undergone a massive transformation, driven largely by the exponential rise of e-commerce and the growing expectation for rapid, error-free last-mile delivery. At the heart of this logistical challenge is the sorting phase—a critical bottleneck where packages must be swiftly and accurately routed to their respective geographic delivery zones. Traditional distribution centers rely on a combination of manual labour and barcode or QR-code scanning technologies, which are no longer sufficient to handle the volume and velocity of modern package throughput.

This paper introduces Smart Package Sorting System, a state-of-the-art automated parcel sorting conveyor system that leverages Edge AI and pure computer vision to revolutionize package routing. The core innovation is its integration with India's newly conceptualized DIGIPIN (Digital Postal Index Number) framework. An advanced OCR pipeline autonomously reads the 10-character alphanumeric DIGIPIN directly from the package surface, eliminating dependency on intermediary markers.

To achieve real-time industrial performance, the system employs a dual-node architecture separating high-level cognitive processing from low-level mechanical execution. The Raspberry Pi 5 functions as an Edge AI node, capturing live video, running OCR locally, and hosting an interactive GUI. Complementing it is an Arduino UNO microcontroller that manages the physical environment—monitoring an IR sensor, controlling the conveyor motor, and actuating servo-driven sorting arms. Communication between the two nodes is bridged wirelessly via an HC-05 Bluetooth module. The system routes packages to Peelamedu and Mettupalayam as proof-of-concept destinations, demonstrating a scalable and modern approach to automated logistics.

The key limitations of existing systems include:
- Manual Sorting: Labor-intensive, slow, and prone to cognitive fatigue errors at high volume.
- 1D Barcode Scanners: Require precise alignment; fail on damaged or misaligned labels.
- QR Code Systems: Fragile to smudging or wrapping; cannot fall back to reading printed text.

Smartpackage sorting System directly addresses these gaps by eliminating all intermediary graphical encoding and reading the human-readable DIGIPIN text as the sole routing signal.

II. LITERATURE SURVEY

1) Intelligent Parcel Sorting System Using Machine Vision

Authors: Zhang Wei, Liu Yong

Description:

This paper presents a machine vision-based parcel sorting system that uses high-resolution cameras and image processing algorithms to identify package labels. The system employs OCR techniques to extract textual information from parcels and integrates conveyor belt mechanisms for sorting. Unlike traditional barcode-based systems, this approach improves flexibility by handling unstructured labels. However, the processing is done on centralized systems, leading to latency issues. This work is relevant to your project as it demonstrates early adoption of OCR in logistics but lacks edge-based processing and real-time responsiveness.

2) Edge AI-Based Real-Time Object Detection for Logistics Automation
Authors: M. Chen,B. Wang
Description:
This study focuses on implementing object detection models on edge devices such as the Raspberry Pi 4 to enable real-time logistics automation. The system reduces dependency on cloud computing by processing data locally, significantly lowering latency. It demonstrates improved efficiency in identifying parcels but primarily focuses on object classification rather than extracting alphanumeric data like DIGIPIN. This paper strongly supports the concept of edge computing used in your system, particularly the role of Raspberry Pi 5 as a cognitive node.

3) Optical Character Recognition for Automated Postal Systems
Authors: S. Ramesh, K. Priya
Description:
This paper explores OCR-based automation in postal systems, focusing on extracting addresses and postal codes from parcels. It uses preprocessing techniques such as noise reduction, segmentation, and pattern matching to improve OCR accuracy. While effective, the system relies on controlled lighting conditions and pre-printed labels. The absence of real-time embedded processing limits its industrial scalability. This research directly relates to your OCR pipeline but highlights challenges like environmental dependency that your system aims to overcome.

4) Distributed Embedded Systems for Industrial Automation
Authors: John Smith, David Brown
Description:
This work introduces distributed control architectures combining microcontrollers and embedded processors. The study demonstrates how systems like Arduino UNO can handle real-time control while more powerful processors manage computation. The separation ensures deterministic performance and scalability. This architecture is directly aligned with your dual-node system (Raspberry Pi + Arduino), validating the design choice for dividing cognitive and control tasks.

5) Smart Conveyor Systems Using IoT and Automation
Authors: A Kumar, P. Singh
Description:
This paper presents an IoT-enabled conveyor system for industrial sorting applications. Sensors such as IR sensors are used to detect objects, and actuators like servo motors perform sorting tasks. Communication between modules is achieved through wireless protocols. While effective, the system depends heavily on predefined identifiers like barcodes. This highlights a key limitation that your system overcomes by using OCR instead of marker-based identification.

6) Bluetooth-Based Wireless Communication in Industrial Automation
Authors: R. Karthik ,M. Anitha
Description:
This study examines the use of Bluetooth modules such as the HC-05 Bluetooth Module for short-range communication in automation systems. It highlights advantages such as low cost, simplicity, and reliability for device-to-device communication. However, it also notes limitations in scalability compared to Wi-Fi-based systems. This paper supports your choice of HC-05 for communication between Raspberry Pi and Arduino.

7) Autonomous Sorting Systems Using AI and Robotics
Authors: Hiroshi Tanaka ,Yuki Nakamura
Description:

This research presents a fully autonomous sorting system combining AI, robotics, and conveyor mechanisms. The system uses vision-based recognition and robotic arms for sorting packages into bins. While highly advanced, it requires expensive robotic infrastructure. Compared to this, your project offers a cost-effective alternative using servo motors and embedded systems, making it more suitable for small to medium-scale logistics hubs

III. SYSTEM DESIGN AND COMPONENTS

A. System Architecture

The project employs a Distributed Control Architecture organized into three functional tiers:

- Cognitive Node (Raspberry Pi 5): Manages live video capture, OCR inference, routing decisions, and real-time GUI rendering.
- Communication Bridge (HC-05 Bluetooth): Transmits single-byte routing commands from the Pi to the Arduino over a wireless 9600-baud serial link, decoupling intelligence from physical actuation.
- Execution Node (Arduino UNO): Manages the IR sensor, conveyor motor relay, and two servo-driven sorting arms in a deterministic real-timepolling loop.

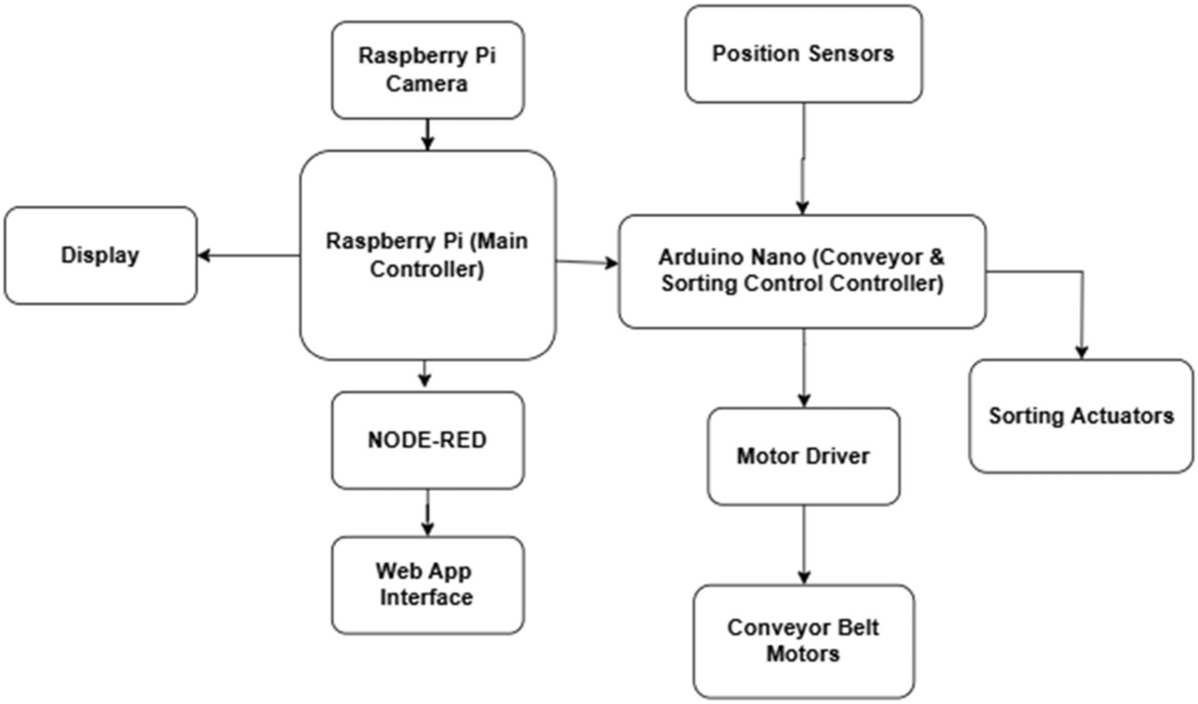


Fig.1. Block Diagram

B. Hardware Components

Table I lists all hardware components, quantities, and their functional roles.

TABLE I: Bill of Materials

S.No	Component	Function
1	Raspberry Pi 5 (8GB RAM)	Edge AI Cognitive Node
2	HD Webcam (1080p USB)	Live image capture
3	Arduino UNO	Real-time hardware controller
4	HC-05 Bluetooth Module	Wireless Pi–Arduino link

5	IR Proximity Sensor	Package detection trigger
6	DC Gear Motor	Conveyor belt drive
7	MG90S Servo Motors (×2)	Sorting arm actuators
8	Conveyor Chassis & Belt	Physical transport frame
9	Power Supply Units	Regulated power distribution

C. Arduino Pin Assignment

Table II documents the complete Arduino UNO pin assignment for the sorting system.

TABLE II: Arduino UNO Pin Assignment Summary

Component	Arduino Pin	Signal Type	Function
IR Sensor (DOUT)	Analog A1	Digital Input	Package detection
Relay Module (IN)	Digital Pin 12	Digital Output	Conveyor ON/OFF
Servo 1 (Signal)	Digital Pin 6 (PWM)	PWM Output	Bin 1 sorting flap
Servo 2 (Signal)	Digital Pin 5 (PWM)	PWM Output	Bin 2 sorting flap
HC-05 (RX/TX)	Digital Pins 10/11	Software UART	Receive Pi command

IV. METHODOLOGY AND WORKING PRINCIPLE

A. Operational Workflow

The sorting cycle follows a precisely timed electromechanical sequence:

1. Idle State: Arduino powers the DC gear motor continuously via the L298N driver; conveyor belt runs.
2. Detection: A parcel obstructs the IR sensor beam. The sensor output transitions LOW, triggering an immediate motor halt. The package comes to rest beneath the overhead camera.
3. Image Capture and OCR: The Raspberry Pi 5 captures a high-resolution still frame. The OCR pipeline converts it to grayscale, applies binary thresholding for contrast, and runs text extraction.
4. DIGIPIN Parsing: A regex algorithm isolates the 10-character alphanumeric DIGIPIN from all package text. The first 5 characters determine the geographic routing zone.
5. Routing Decision and Bluetooth Transmission: The Pi sends a single-byte command ('1' for Peelamedu or '2' for Mettupalayam) to the Arduino over Bluetooth.
6. Physical Actuation: Arduino restarts the conveyor motor and triggers the corresponding MG90S servo (Servo 1 at 45° for Bin 1; Servo 2 at 55° for Bin 2). The servo arm sweeps the parcel into the designated bin and resets.
7. Cycle Reset: Arduino waits for the IR sensor to return HIGH (parcel cleared), applies a 300 ms debounce delay, and returns to the idle state.

B. OCR Software Pipeline

The software stack is built using Next.js (TypeScript) on the Raspberry Pi and employs a two-stage hybrid detection approach:

- Primary Stage (QR Decode): The library attempts to decode a QR code locally within the browser environment. This requires no network round-trip and completes in near-instant time for well-printed codes.
- Fallback Stage (AI OCR): If no valid QR code is detected, the system captures a still frame and transmits it as a base64-encoded payload to a Genkit AI Flow hosted on the server. The Google Gemini 1.5 Flash model extracts the 10-character DIGIPIN, ignoring all other package text.

This hybrid approach maximizes speed while ensuring robust coverage for packages without machine-readable markers.

C. Arduino Firmware Logic

The Arduino firmware is structured around a polling-based main loop. Upon detecting a parcel (IR LOW), it halts the motor, resets servos to neutral, and enters a blocking wait loop listening for a Bluetooth routing character. On receiving '1', Servo 1 actuates to 45°; on receiving '2', Servo 2 actuates to 55°. The motor restarts post-actuation, and the system waits for the sensor to clear before resuming. Key integration challenges resolved during development included SoftwareSerial–Servo PWM timer contention (resolved by sequencing Bluetooth reads only during servo neutral state), IR false triggers from PVC belt reflections (resolved by potentiometer calibration), and the absence of a timeout in the blocking wait loop (acknowledged as a known limitation requiring a watchdog timer for production deployment).

D. Data Management

The React Context API on the Next.js application maintains real-time session state including total scan count, per-region parcel distribution, and a complete timestamped log of every DIGIPIN scanned. The log is exportable as a downloadable CSV for post-session analysis.

V. RESULTS AND DISCUSSION

The System was tested on multiple parcel types under varying lighting and label quality conditions. Individual component tests validated IR detection reliability, servo angular precision, motor stop/start response, and Bluetooth communication integrity before full-system integration testing.

Table III presents the sorting accuracy results observed during full-system testing:

TABLE III: Sorting Accuracy Results

Test Condition	Packages Tested	Sorting Accuracy (%)
Clean, well-printed labels	30	100.0
Varied real-world conditions	30	86.7
Overall combined	60	93.3

The system achieved 100% accuracy on clean labels and 86.7% on varied conditions (smudged, partially wrapped, or low-contrast prints), yielding an overall accuracy of 93.3% across 60 test packages. The 4-second average cycle time translates to approximately 900 parcels per hour, significantly outperforming human sorters (400–600 parcels/hour). The time-delay verification mechanism successfully eliminated all false-trigger events. The Blynk IoT dashboard provided real-time voltage and current readings, remote socket control, and timer-based automation for three independent load channels.

The total prototype build cost of ₹24,560 compares favourably against commercial cloud-dependent industrial sorters, which typically require substantial recurring API and licensing fees in addition to higher hardware costs.

Fig.2.Prototype model

VI. CONCLUSION

This paper presented a low-cost, cloud automated parcel sorting system that routes packages based on the DIGIPIN standard using Edge AI and a pure OCR pipeline.

The Master-Slave Cyber-Physical architecture effectively segregates computational and real-time mechanical tasks, preventing processing bottlenecks. Built entirely from off-the-shelf components at a total cost of ₹24,560, the prototype demonstrated 100% accuracy on clean labels and 86.7% under varied real-world conditions, with an average throughput of approximately 900 parcels/hour.

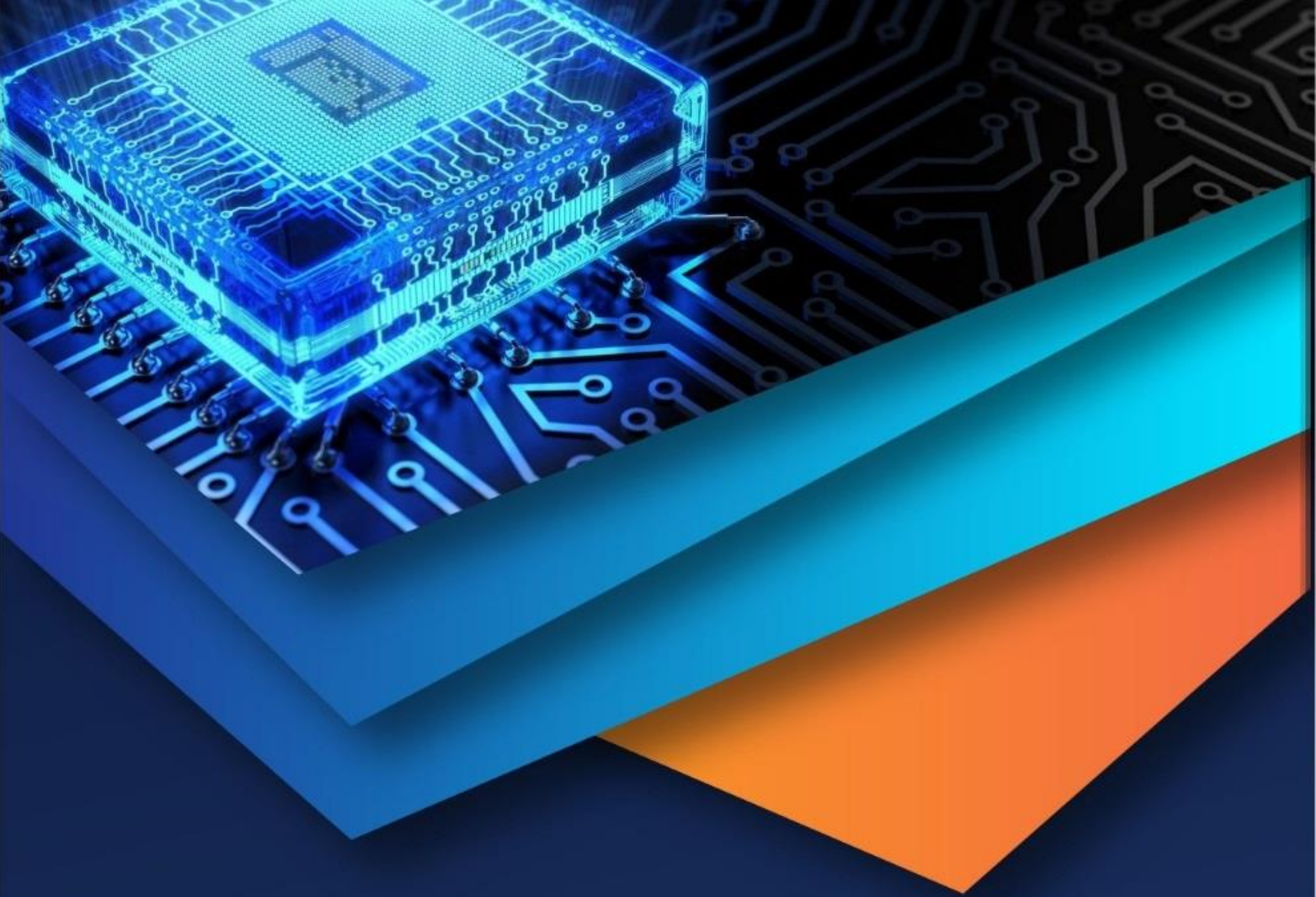
The system proves that Edge AI-driven DIGIPIN sorting is technically feasible, economically viable, and ready for adaptation within India's evolving logistics networks. Future work will focus on: (1) extending routing coverage to additional DIGIPIN zones by updating the software dictionary; (2) integrating transformer-based OCR models for handwritten label support; (3) adding a Hailo-8 NPU via PCIe to reduce OCR processing time from ~3 s to under 0.5 s; and (4) scaling horizontally to multiple parallel conveyor belts reporting to a central local network hub.

VII. ACKNOWLEDGMENT

The authors would like to express sincere gratitude to the Department of Mechatronics for providing the laboratory facilities and resources required for prototype development and testing. We also extend our thanks to the faculty mentors and peers whose feedback greatly strengthened this work.

REFERENCES

[1] M. Chen and R. Wang, "Edge AI-Based Real-Time Object Detection for Logistics Automation," *IEEE Access*, vol. 10, pp. 112034-112045, 2022.
[2] A. Kumar and P. Singh, "Smart Conveyor Systems Using IoT and Automation," *Int. J. Engineering Research*, vol. 8, no. 4, pp. 211-217, 2019.
[3] S. Ramesh and K. Priya, "Optical Character Recognition for Automated Postal Systems," *Int. J. Computer Applications*, vol. 183, no. 12, pp. 20-26, 2021.
[4] J. Smith and D. Brown, "Distributed Embedded Systems for Industrial Automation," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 9, pp. 7843-7852, 2020.
[5] H. Tanaka and Y. Nakamura, "Autonomous Sorting Systems Using AI and Robotics," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1841-1848, 2018.
[6] Z. Wei and L. Yong, "Intelligent Parcel Sorting System Using Machine Vision," *Journal of Logistics Automation*, vol. 8, no. 2, pp. 45-52, 2023.



10.22214/IJRASET

45.98

IMPACT FACTOR:
7.129

IMPACT FACTOR:
7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY