International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

IOT Smart Waste Management System with AI and Analytics

Dr.PRATIBHADEVI TAPASHETTI [1], BHUMIKA M S [2], AMAR [3] , BHAGYASHRI R K[4] , TEJASWINI J [5]

[1]ECE , AMRUTA INSTITUTE OF ENGINEEROING AND MANAGEMENT SCIENCES , BIDADI , BENGALURU

[2]ECE , AMRUTA INSTITUTE OF ENGINEEROING AND MANAGEMENT SCIENCES , BIDADI , BENGALURU

[3] ECE , AMRUTA INSTITUTE OF ENGINEEROING AND MANAGEMENT SCIENCES , BIDADI , BENGALURU

[4]ECE , AMRUTA INSTITUTE OF ENGINEEROING AND MANAGEMENT SCIENCES , BIDADI ,BENGALURU

[5]ECE , AMRUTA INSTITUTE OF ENGINEEROING AND MANAGEMENT SCIENCES , BIDADI , BENGALURU

Abstract -This paper proposes an Internet of Things (IoT)-based smart waste management prototype integrating multi-point bin fill-level sensing, weight measurement, camera-assisted waste classification, automated sorting, and data analytics. An ESP32-S3 controller gathers measurements from four VL53L0X time-of-flight distance sensors through a TCA9548A I²C multiplexer and from a load cell through an HX711 interface. A compatible AI camera supplies images to a trained classifier, while a stepper motor, driver, servo, and limit switch support controlled sorting experiments. Time-stamped readings can be sent over Wi-Fi to a dashboard for status monitoring, threshold alerts, and historical analysis. The paper presents the proposed architecture, hardware interfaces, processing logic, and an experimental protocol for evaluating sensing accuracy, classification performance, communication reliability, and sorting success. No experimental results are claimed because validated prototype measurements have not yet been provided. The work therefore serves as an implementation and evaluation framework that should be completed with measured data before performance conclusions are drawn.

Key Words: Internet of Things, smart waste management, ESP32-S3, waste classification, data analytics, automated segregation.

1. INTRODUCTION

Solid waste management depends on timely information about bin occupancy, waste composition, and collection demand. Fixed collection schedules and manual inspection provide limited visibility into the current condition of individual bins. Depending on location and usage, this can lead to overflowing bins or collection visits before a bin is full. Internet of Things (IoT) systems can provide continuous or periodic measurements from distributed bins, while data analytics can turn those measurements into information for operators

[1]. Artificial intelligence (AI), particularly image-based classification, can support waste identification and automated sorting. Its performance depends on training data, image quality, lighting, object overlap, and available processing capability

[2]. An integrated design must therefore consider sensing, communications, model deployment, actuation, and power distribution together.

This paper proposes a prototype architecture based on an ESP32-S3 controller, multiple VL53L0X distance sensors, a load cell, an AI camera, and a motorized sorting mechanism. The contribution is a component-level integration plan and a reproducible evaluation protocol. Since measured prototype data have not been supplied, this manuscript does not claim

achieved accuracy, energy savings, or collection-cost reductions.

2. RELATED WORK

Reviews of IoT-enabled smart waste management identify sensing, information exchange, collection services, and route planning as recurring system-level themes [1]. Effective systems need to translate sensor readings into useful operational information rather than only displaying raw measurements.

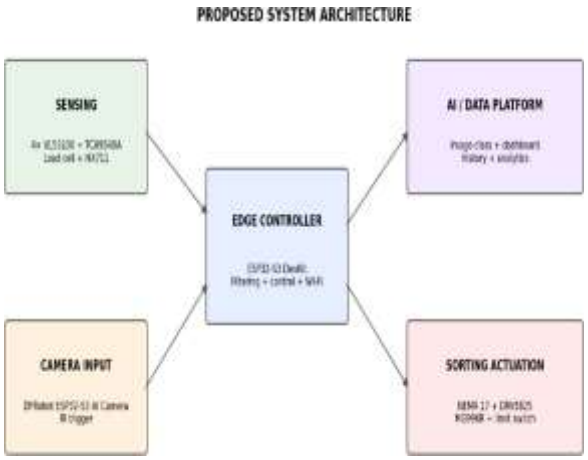


Fig. 1: Proposed architecture of the smart waste management system

AI-based waste classification research discusses image datasets, convolutional neural networks, transfer learning, and deployment challenges [2]. Real-world waste may be dirty, crushed, partially hidden, or visually similar to other materials. A practical classifier should therefore include an unknown or manual-review outcome for low-confidence predictions and should be evaluated on data separate from training images.

The proposed design combines fill-level and weight measurements with camera-assisted classification and physical sorting. Its value must be evaluated against simpler baselines, such as sensor-only monitoring or manual sorting

3. PROPOSED SYSTEM ARCHITECTURE

The system contains five functional layers: sensing, edge control, image classification, actuation, and data storage/analytics. Figure 1 presents the proposed data flow. The sensing layer estimates bin occupancy and weight. The ESP32-S3 gathers readings and transmits status records over Wi-Fi. The AI camera provides images to a compatible classification model. When a valid classification and safe operating conditions are available, the controller can command a sorting gate or platform. A dashboard stores time-stamped records and displays status, alerts, and trends

The selected hardware includes an ESP32-S3-DevKitC-1 N32R16V board, a DFRobot ESP32-S3 AI Camera Module, four VL53L0X distance sensors, a TCA9548A I²C multiplexer, a 20 kg load cell with HX711, an inductive metal proximity sensor, a capacitive moisture sensor, an IR sensor, a WS2812B addressable LED ring, a NEMA 17 stepper motor with DRV8825 driver, an MG996R servo, and a mechanical limit switch. The role of each sensor must be defined during implementation. For example, a capacitive moisture sensor requires calibration and should not be treated as a universal waste-material identifier.

4. HARDWARE DESIGN AND INTERFACING

4.1 Fill-level sensing: Four VL53L0X time-of-flight sensors can be positioned at selected points inside or above a bin to measure the distance to the waste surface. Since these sensors normally share a default I²C address, the TCA9548A multiplexer can isolate them on separate channels. Invalid readings should be rejected, and valid readings may be filtered or combined to reduce the effect of an uneven waste surface.

4.2 Weight measurement: The load cell connects to the HX711 amplifier and converter. The cell should be mounted so that the intended load passes through it without side loading. Calibration with known reference masses and a tare procedure are required. Weight complements distance-based fill estimation but does not directly identify material type.

4.3 Camera and classification: The exact DFRobot camera revision, supported firmware, model format, and inference location must be confirmed. If the camera/controller combination cannot perform inference locally, images may be sent to a local computer or server. The model should be

trained or fine-tuned on representative images and evaluated using a held-out test set.

4.4 Sorting mechanism: The NEMA 17 stepper motor, DRV8825 driver, MG996R servo, and mechanical limit switch can position a platform, rotate a chute, or operate a gate. The mechanical design must define the motion, waste path, compartments, end stops, and jam response. The driver current limit, cooling, servo stall current, and mechanical load must be checked before sustained operation.

4.5 Power and electrical safety: The Mean Well LRS-150-12 provides a nominal 12 V DC rail. MINI560 step-down modules can provide regulated 5 V where their current capacity and thermal performance are sufficient. Include the XL4016E1 only if a defined load requires it. Never connect 12 V directly to ESP32 GPIO, 3.3 V, or 5 V logic pins. Verify every module's voltage tolerance and use level shifting where needed. Provide appropriate fusing, wiring, strain relief, ventilation, and an enclosure. Mains wiring and protective earthing should be handled by a qualified person. Plan separate motor and logic power paths to reduce resets caused by current transients.

5. SOFTWARE AND DATA PROCESSING

5.1 Fill percentage: Let H be the calibrated usable bin height and d the measured distance from the sensor reference plane to the waste surface. A basic estimate is $F = 100 \times (H - d) / H$, clipped to 0–100%. For multiple sensors, valid readings may be combined using a median or a calibrated weighted estimate. This calculation is valid only when mounting geometry and reference height are known.

5.2 Data records: Each record should contain a bin identifier, timestamp, sensor distances, estimated fill percentage, measured weight, sensor-validity flags, predicted class and confidence (if available), actuator state, and network status. Data can be transmitted to a database and dashboard using an API or messaging protocol. Local buffering and retry logic should prevent temporary network loss from blocking control tasks.

5.3 Classification confidence: The classifier returns a predicted category and confidence score. A configurable threshold can route uncertain images to an unknown/manual-review state instead of triggering a potentially incorrect sort. Training, validation, and test images should be separated to reduce data leakage. Report per-class precision and recall in addition to overall accuracy, particularly when classes are imbalanced.

5.4 Predictive analytics: Historical fill records can estimate the fill rate and predict when a bin may cross a collection threshold. A simple baseline is time-to-threshold $= (T - F) / r$, where T is the target fill percentage, F is current fill, and r is a positive estimated fill rate. The estimate is not valid when r is zero or negative and may fail when usage changes suddenly. More advanced forecasting should be adopted only after enough representative data have been collected.

5.5 Sorting control: A sorting cycle should begin only when an object trigger is valid, classification confidence is sufficient, the target compartment is available, and the actuator state is known. Firmware should enforce motion timeouts, monitor the limit switch, and enter a recovery state after a jam or unexpected movement. Initial tests should avoid hazardous or sharp waste.

6. EXPERIMENTAL METHODOLOGY

The prototype should be evaluated in a controlled environment before performance claims are made. For fill-level sensing, collect readings at multiple known heights and surface shapes, compare estimates with reference measurements, and report mean absolute error (MAE), root mean square error (RMSE), maximum absolute error, and invalid-reading rate.

For weight measurement, calibrate using known masses across the intended range and report MAE, repeatability, and zero drift after tare. For AI classification, create a labelled dataset covering selected categories, lighting, orientation, occlusion, and background variation. Use separate training, validation, and held-out test sets; report accuracy, per-class precision and recall, macro-F1, a confusion matrix, inference time, and the proportion of low-confidence predictions.

For communication reliability, report message delivery ratio, end-to-end latency, reconnection time, and data loss during network interruptions. For sorting, conduct repeated trials for each supported class and report routing success, mis-sorting rate, cycle time, jam frequency, and safe-stop response. For prediction or collection planning, compare

forecasts and routes against a stated baseline using forecast error, travel distance, collection time, and overflow events. Table 1 summarizes the proposed evaluation plan. Sample sizes, test duration, thresholds, and results must be completed after implementation.

Table 1: Proposed experimental evaluation plan.

Subsystem	Metrics	Evaluation method
Fill-level sensing	MAE, RMSE, invalid readings	Compare estimates with reference heights
Weight sensing	MAE, repeatability, zero drift	Known masses and repeated loading
AI classification	Accuracy, precision, recall, macro-F1	Held-out labelled image set
Connectivity	Delivery ratio, latency, recovery	Timestamped message logs
Sorting	Routing success, cycle time, jam rate	Repeated trials by waste class
Forecasting	Forecast error, route distance, overflow	Compare with stated baseline

7. RESULTS AND DISCUSSION

No measured prototype results are reported because experimental logs, labelled test images, calibration data, and sorting trials have not been provided. Reporting invented values would make the evaluation irreproducible. The current paper should therefore be treated as a proposed system and evaluation framework until the experiments in Section 6 are completed.

Performance may be affected by sensor angle, waste-surface geometry, dust, obstructions, load-cell mounting, vibration, image lighting, occlusion, unfamiliar materials, network outages, and actuator current spikes. These factors should be documented alongside quantitative results.

The integration of multiple measurements and functions is a potential benefit, but integration alone does not prove better collection outcomes. Claims about reduced overflow, fewer vehicle trips, cost savings, or environmental impact require comparison against a defined baseline and measurements over an appropriate period.

8. CONCLUSION

This paper presented a proposed IoT-based smart waste management system combining multi-point fill-level sensing, weight measurement, camera-assisted waste classification, motorized sorting, and data analytics. The architecture uses an ESP32-S3 controller and TCA9548A multiplexer for multiple VL53L0X sensors, an HX711 interface for load-cell readings, and a motor/servo arrangement for sorting experiments. A dashboard can present current status and historical trends, while future records may support fill-level forecasting and collection prioritization.

The design is a prototype framework, not proof of achieved performance. Sensor calibration, model evaluation, actuator safety, and baseline comparisons are required before claims of accuracy or operational improvement can be supported. A final submission should include measured results, test conditions, implementation details, and limitations.

ACKNOWLEDGEMENT

The authors may acknowledge institutional support, laboratory facilities, or project guidance here, if applicable. Remove this sentence if no acknowledgement is needed.

REFERENCES

[1] I. Sosunova and J. Porras, “IoT-Enabled Smart Waste Management Systems for Smart Cities: A Systematic Review,” IEEE Access, vol. 10, pp. 73326–73363, 2022, doi: 10.1109/ACCESS.2022.3188308.

[2] D. V. Yevle and P. S. Mann, “Artificial intelligence based classification for waste management: A survey based on taxonomy, classification & future direction,” Computer Science Review, vol. 56, Art. no. 100723, 2025, doi: 10.1016/j.cosrev.2024.100723.

[3] Espressif Systems, “ESP32-S3 Series Datasheet” and official ESP-IDF/Arduino documentation. Consult the documentation for the exact board revision and software framework used.

[4] STMicroelectronics, “VL53L0X Time-of-Flight Ranging Sensor” datasheet. Consult the current official datasheet and the documentation for the breakout board used.

[5] Texas Instruments, “DRV8825 Stepper Motor Controller IC” datasheet. Set current limit and cooling according to the driver board and motor specifications.

[6] DFRobot, “ESP32-S3 AI Camera Module” product and software documentation. Confirm supported firmware, model format, and inference workflow for the exact module revision.