International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 5, Sep-Oct-2026, ISSN (Online): 2395-566X

Revolutionizing Food Safety Through AI-Powered Freshness Detection and Intelligent Expiry Prediction

Gunjan and Rishita Vohra[1], Mr.Ritiesh Kumar Chandel[2], Dr. Akhilesh Das[3]
Department of Artificial Intelligence and Data Science[1,2]
Institute of Technology & Management (formerly Northern India Engineering College) Affiliated to Guru Gobind Singh Indraprastha University, Delhi, India[3]

Abstract — Food spoilage and the premature disposal of edible food are persistent problems for public health, the economy and the environment. Fixed best-before and expiry dates are set under controlled conditions and ignore the temperature fluctuations and handling that food experiences between farm and fridge. This paper reviews recent artificial intelligence (AI) approaches to freshness detection and shelf-life prediction, covering convolutional neural network (CNN) image classifiers and low-cost gas-sensor systems with machine-learning models, and compares their reported performance. Building on the gaps identified, we formulate the problem mathematically and propose a four-layer vision–sensor fusion framework that combines a transfer-learned CNN with a regression model to estimate remaining shelf life, together with a confidence-aware decision rule that sends uncertain items to manual inspection. We also describe an edge-deployment architecture, a planned experimental protocol and the main risks. Reported accuracies in the reviewed sensor-based studies range from 91.9% to 97.05%. The framework is conceptual; experimental validation on a multi-product dataset is identified as future work.

Keywords— food safety, freshness detection, expiry prediction, shelf life, convolutional neural network, transfer learning, electronic nose, Internet of Things, edge computing, machine learning..

I. INTRODUCTION

Food spoilage is a food-safety problem as well as an economic and environmental one. The Food and Agriculture Organization estimates that roughly one-third of the food produced for human consumption is lost or wasted globally, about 1.3 billion tonnes a year [1]. A significant share of the waste at the consumer end is edible food discarded because its best-before date has passed [2]. Printed dates include large safety margins and are determined under controlled laboratory conditions. During farm-to-fridge transport, food is exposed to mishandling and temperature fluctuations, so the printed date no longer reflects the real condition of the product; one survey reports that 37% of users discard food every time it passes its expiry date [3]. The outcome is a double failure: safe food is wasted, while food that spoiled early because of poor storage can still reach the consumer.

Artificial intelligence (AI) offers a way to replace static labels with item-level assessment. Deep learning, and convolutional neural networks (CNNs) in particular, can learn visual cues of freshness such as colour, texture and surface defects directly from images without hand-crafted features [4], [5]. In parallel, low-cost temperature, humidity and gas sensors combined with machine-learning (ML) models can sense the volatile compounds released during spoilage and estimate the remaining shelf life [3], [2], [6]. These two lines of work have largely developed separately, and each answers only part of the question a consumer or retailer asks: is this item safe now, and for how many more days will it stay safe?

The contributions of this paper are: (i) a structured review of vision-based and sensor-based approaches to freshness detection and expiry prediction; (ii) a taxonomy and comparison of the reported methods; (iii) a mathematical formulation of the freshness-and-expiry problem that makes the cost of unsafe errors explicit; (iv) a four-layer vision–sensor fusion framework with a confidence-aware decision rule and an edge-deployment architecture; and (v) a planned experimental protocol with an analysis of risks. The study is literature-based; all performance figures quoted are those reported by the cited authors, and no new experimental results are claimed.

The rest of the paper is organised as follows. Section II gives the background on spoilage and shelf life. Section III reviews related work.

Section IV formulates the problem. Section V presents the proposed framework. Section VI describes the experimental design. Section VII compares and discusses the approaches, Section VIII covers challenges and threats to validity, Section IX outlines future work, and Section X concludes.

© 2026 IJSRET

Background

Food Spoilage and Shelf Life

Shelf life is the period during which a food stays safe and keeps acceptable quality under stated storage conditions. Quality is lost through microbial growth, enzymatic activity and chemical changes such as oxidation, and the speed of these processes depends strongly on temperature and humidity. Temperature plays a central role because it directly influences these biochemical activities [6]. A common simplification is that the rate of quality loss rises by a fixed factor (often written Q_{10}) for every 10 °C increase in temperature. Fig. 1 illustrates what this means for a printed date: the same product reaches its acceptability limit much earlier in warm storage and much later in a cold chain.

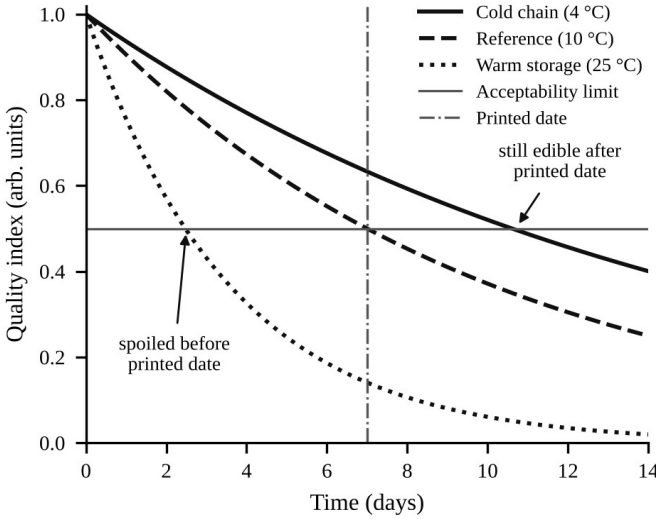


Fig. 1. Illustrative quality decay at three storage temperatures with $Q_{10} = 2$ (arbitrary units, not measured data). A single printed date is unsafe for warm storage and wasteful for cold storage.

2. Limits of Printed Dates

A printed date is one number computed for an assumed storage history. It cannot know whether a particular item sat in a hot truck or a well-run cold room. Fig. 1 shows both failure modes: items that spoil before the printed date, which is a safety risk, and items that remain edible well after it, which causes waste. Both outcomes motivate item-level assessment based on observed condition and actual storage environment.

3. Observable Spoilage Indicators

Spoilage leaves measurable traces. Visible changes include colour shifts such as browning or loss of green, surface wrinkling, softening and mould growth; these are what a camera can capture. Chemical changes release volatile organic compounds whose mix depends on the food: fermentation of fruit sugars tends to produce alcohols and esters, protein-rich foods tend to release nitrogen- and sulphur-containing compounds, and respiring produce changes the carbon dioxide level in its surroundings. Gas sensors respond to such changes even before a visible sign appears [3], [2]. Temperature and humidity do not measure spoilage itself but explain how fast it proceeds. A practical system therefore benefits from combining appearance, volatile signature and storage conditions.

4. Technology Enablers

Three developments make item-level assessment practical. First, CNNs trained on large image collections such as ImageNet [8] learn general visual features that transfer to new tasks, and transfer learning reduces the labelled data needed for a specialised task [7]. Second, low-cost gas, temperature and humidity sensors can be read by inexpensive microcontrollers [3]. Third, TinyML and tiny deep learning allow compact models to run on resource-constrained devices, which removes the need for a constant network link [6], [9]. Together these allow freshness assessment close to where food is stored, in cold rooms, shops and homes.

III. RELATED WORK

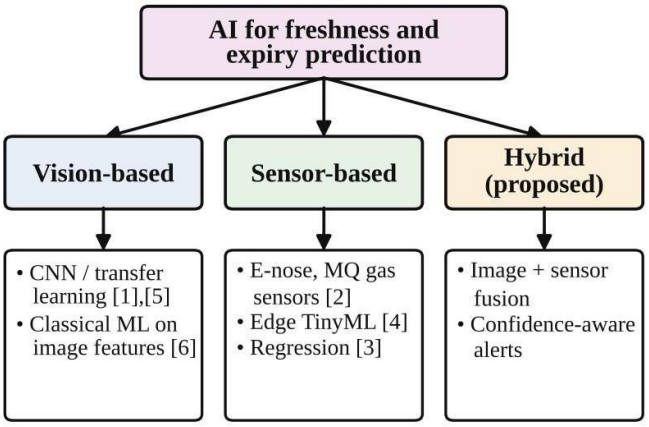


Fig. 2. Taxonomy of AI approaches to freshness detection and expiry prediction.

1. Vision-Based Freshness Detection

Wang et al. review CNN-based detection across the whole fresh-fruit production chain, from flower detection through fruit detection and picking to grading, and report that CNN models improved to suit each stage give the best detection

© 2026 IJSRET

results [4]. Marcuzzo et al. survey fruit ripeness classification and note that classical ML methods such as support vector machines, decision trees and k-nearest neighbours were applied successfully before deep learning methods such as CNNs showed very promising results [5]. Gautam and Chaurasia study CNN-based forecasting of fruit stock life, point out that manual quality supervision varies with the inspector, and discuss ResNet50, MobileNetV2, DenseNet-121, NASNet-A and EfficientNet with transfer learning [10].

2. Sensor-Based Spoilage and Expiry Prediction

Nair et al. built a low-cost portable electronic nose from an MQ-2 gas sensor with temperature and humidity sensors, using an ESP8266 and Arduino for data acquisition. Logistic regression, decision tree, support vector and k-nearest neighbour classifiers trained on odour data from bananas and tomatoes reached 97.05% accuracy in deciding whether the fruit had gone stale [3]. Wunderlich et al. showed that a VOC or carbon dioxide sensor combined with a random forest or extreme gradient boosting regressor can accurately predict the day of spoilage of fresh pizza, a cheaper and more precise alternative to conventional best-before determination [2]. A recent IoT study on dates combined a multichannel gas sensor with a TinyML model on an edge device and obtained 91.9% classification accuracy with an AUC of 0.98; the authors note that most earlier IoT systems only collect environmental data and do not estimate shelf life directly [6].

3. Spectral Imaging and Supply-Chain Systems

Beyond ordinary cameras, a recent review covers hyperspectral imaging combined with deep learning for the quality and safety inspection of fruits and vegetables [11]. Such systems capture information outside the visible range but need costly hardware. At the system level, IoT-based frameworks for fresh produce supply chains under uncertainty have also been proposed [12], which shows that item-level freshness data can feed wider logistics decisions.

4. Summary and Research Gaps

Table I summarises the reviewed work. Three gaps emerge. First, vision-based studies mostly classify the present state (fresh or rotten) and rarely predict how long the food will remain safe.

Second, sensor-based studies are usually specific to one product and use small datasets, which limits generalisation. Third, few works fuse image and sensor data in one pipeline that can run on low-cost edge hardware and that treats unsafe errors differently from harmless ones.

Table 1: Comparison of Reviewed Approaches

Study	Technique / input	Reported outcome
Wang et al. [4]	CNN-based detection across fruit production stages (review)	Stage-specific improved CNNs perform best
Marcuzzo et al. [5]	Survey of ripeness classification: SVM, trees, KNN, ANN, CNN	CNNs show very promising results
Gautam & Chaurasia [10]	ResNet50, MobileNetV2, DenseNet-121, EfficientNet, SVM; images	Comparative study; no single headline figure
Nair et al. [3]	MQ-2 gas, temperature, humidity; LR, DT, SVC, KNN	97.05% accuracy (stale vs. fresh)
Wunderlich et al. [2]	VOC / CO_2 sensor; random forest, XGBoost regression	Accurate prediction of spoilage day
Dates estimator [6]	Multichannel gas sensor; TinyML on edge device	91.9% accuracy, AUC 0.98
Proposed (this paper)	CNN + sensor features + regression, confidence-aware	To be validated experimentally

5. Comparison of Sensing Modalities

The reviewed work also differs in what is sensed. Table II contrasts the main modalities. Images are cheap and contact-free but see only the surface; gas sensors detect chemical change earlier but drift and respond to several compounds at once; hyperspectral imaging adds information beyond the visible range at a much higher hardware cost [11]. The complementary strengths and weaknesses are the main argument for fusion.

Table 2: Strengths and Weaknesses of Sensing Modalities

Modality	Strengths	Weaknesses
RGB image + CNN	Non-invasive; cheap; no contact with food	Sees surface only; sensitive to lighting and occlusion
Gas / VOC sensors	Detects chemical change early; low cost	Drift; cross-sensitivity; product-specific

© 2026 IJSRET

Hyperspectral imaging [11]	Information beyond the visible range	Costly hardware; large data volumes
Fusion (proposed)	Complementary signals; fallback modes	More complex; needs paired training data

IV. PROBLEM FORMULATION

Let an item be described by an image x and a sensor window s (gas, temperature and humidity readings over a recent period), and let c denote the product type. The goal is to produce (a) a freshness class y with confidence p, and (b) the remaining shelf life d in days. Quality loss is modelled as first-order decay whose rate depends on temperature through a Q10 relation:

$$k(T) = k\text{ref} \cdot Q10\ (\text{T} - \text{T_ref})/10 \qquad (1)$$

$$q(t) = q0\ \exp(-k\ t) \qquad (2)$$

where q is a quality index and qmin the acceptability limit. The remaining shelf life of an item with current quality q stored at a constant temperature is then

$$d = \ln(\ q\ /\ q\text{min}\ )\ /\ k(T) \qquad (3)$$

In practice neither the quality index nor the rate constant is observed directly, so the framework learns d from image and sensor features instead of assuming the decay law. The law is useful as a guide: it shows why temperature history must be an input and why a printed date cannot be correct for every storage condition. Because the consequences of errors are unequal, the decision rule minimises an expected cost rather than the error rate:

$$C = c\text{FN} \cdot FN + c\text{FP} \cdot FP \quad (4)$$

with cFN ≫ cFP, since passing a spoiled item as fresh is a safety risk while discarding a good item is a smaller loss. This asymmetry motivates the confidence threshold and manual-inspection route in Section V-D.

V. PROPOSED FRAMEWORK

1. System Architecture

The framework has four layers (Fig. 3). Layer 1 acquires a camera image together with temperature, humidity and VOC or gas readings. Image and sensor streams are preprocessed separately. Layer 2 runs a CNN that outputs a freshness class and a confidence score, and a sensor trend model that captures the spoilage signature. Layer 3 fuses these outputs in a regression model that estimates the remaining shelf life in days. Layer 4 presents the result on a mobile or web dashboard and raises alerts. The separation of layers lets each part be replaced or retrained independently, for example swapping the CNN backbone without touching the regressor.

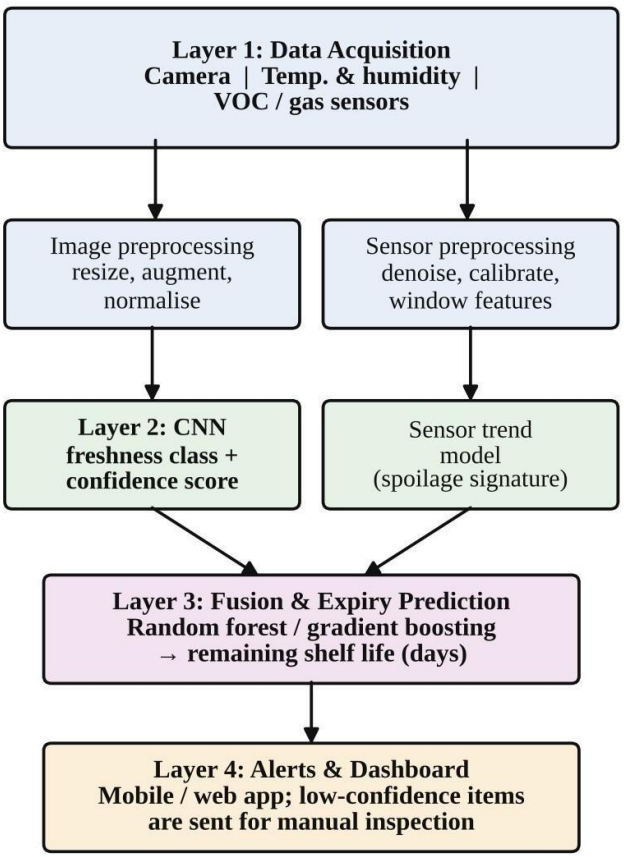


Fig. 3. Proposed four-layer vision–sensor fusion framework.

2. Freshness Classification Module

Input images are resized to 224 × 224 pixels. A backbone initialised with ImageNet weights is adapted by transfer learning (Fig. 4): the early blocks, which encode edges, colour and texture, are kept frozen, while the last convolutional stages are fine-tuned to learn freshness-specific features such as browning, wrinkling and mould. Global average pooling, a dropout layer and a softmax layer then produce the class probabilities and the confidence p. Three candidate backbones are considered (Table III): ResNet-50 [13] as an accuracy baseline, MobileNetV2 [14] for edge deployment because of its small size, and EfficientNet-B0 [15] as a balance between accuracy and size.

© 2026 IJSRET

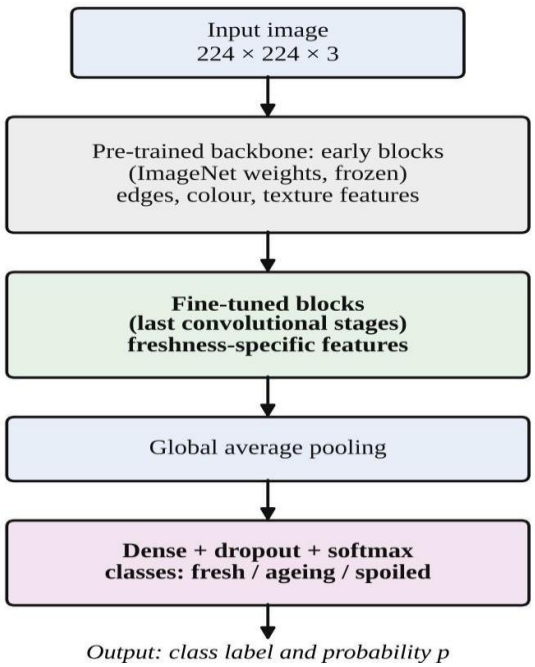


Fig. 4. CNN with transfer learning: frozen early blocks, fine-tuned late blocks and a softmax output.

The network is trained by minimising the cross-entropy between the true class and the predicted probabilities. For K freshness classes, the softmax turns the final scores zk into probabilities pk, and the loss for one image with one-hot label yk is

$$p\mathrm{k} = \exp(z\mathrm{k}) / \Sigma\mathrm{j}\ \exp(z\mathrm{j}) \quad (5)$$

$$L = -\ \Sigma\mathrm{k}\ y\mathrm{k} \log p\mathrm{k} \quad (6)$$

The confidence used by the decision rule is the largest pk. Deep networks are often over-confident, so calibration is checked (Section VI-D) and, if needed, a post-hoc recalibration step such as temperature scaling is fitted on the validation runs.

Table 3: Candidate Cnn Backbones (Values as Reported in the Original Papers)

Backbone	Parameters	ImageNet top-1	Intended role
ResNet-50 [13]	≈26 M	≈76.0%	Accuracy baseline
MobileNetV2 [14]	≈3.4 M	≈72.0%	Edge deployment
EfficientNet-B0 [15]	≈5.3 M	≈77.1%	Accuracy–size balance

3. Expiry Prediction Module

For each item a feature vector is built from windowed sensor statistics (mean, slope and maximum of the gas, temperature and humidity signals), the CNN probability p, and the product type. A random forest [16] or gradient-boosting [17] regressor, which performed well for spoilage-day prediction in [2], maps this vector to the remaining shelf life d in days. Training labels are the observed number of days until spoilage in controlled storage runs. Tree-based regressors suit this task because the data are small and tabular and because they need far less computing than a neural network. The sensor channels used and their roles are listed in Table IV.

Table 4: Sensing Channels and Their Roles

Sensor	Measures	Role in framework	Used in
MQ-series gas sensor	Combustible / volatile gases	Spoilage signatur e	[3]
VOC / CO_2 sensor	Volatile organics, carbon dioxide	Spoilage-day regression	[2]
Multichannel gas sensor	Several gases at once	Edge shelf-life estimate	[6]
Temp. / humidity sensor	Storage conditions	Rate correction, features	[3], [6]
RGB camera	Visual appearance	CNN freshness class	[4], [10]

The window statistics fed to the regressor are kept deliberately simple so that they can be computed on a small device (Table V). Mean level, slope and peak value describe the volatile signature; the temperature and humidity summary lets the model correct for storage conditions, as suggested by Eq. (1); the CNN probability carries the visual evidence; and product type and storage time give context. Keeping the feature set small also makes the model easier to inspect, since the importance of each feature can be read directly from a tree-based regressor.

Table 5: Features Used by the Expiry Regressor

Feature	Description	Purpose
Mean level	Average gas reading in the window	Baseline condition
Slope	Rate of change of the gas reading	Speed of spoilage onset

© 2026 IJSRET

Peak value	Maximum reading in the window	Bursts of volatile release
Temp. / RH summary	Mean and variation	Rate correction (Eq. 1)
CNN probability p	Freshness confidence from the image	Visual evidence
Product type c	Categorical product code	Product-specific behaviour
Storage time	Hours since logging started	Elapsed-time context

4. Confidence-Aware Decision Logic

Because a missed spoiled item is a safety risk, the system does not act on every prediction (Fig. 5). If the CNN confidence is below a threshold τ, the item is sent for manual inspection. Otherwise the regressor predicts the remaining days d. If d exceeds a minimum dmin the predicted expiry date is shown as safe; if not, a use-soon or discard alert is raised. The threshold τ is tuned on a validation set to reach a target recall on the spoiled class, which implements the cost asymmetry of Eq. (4).

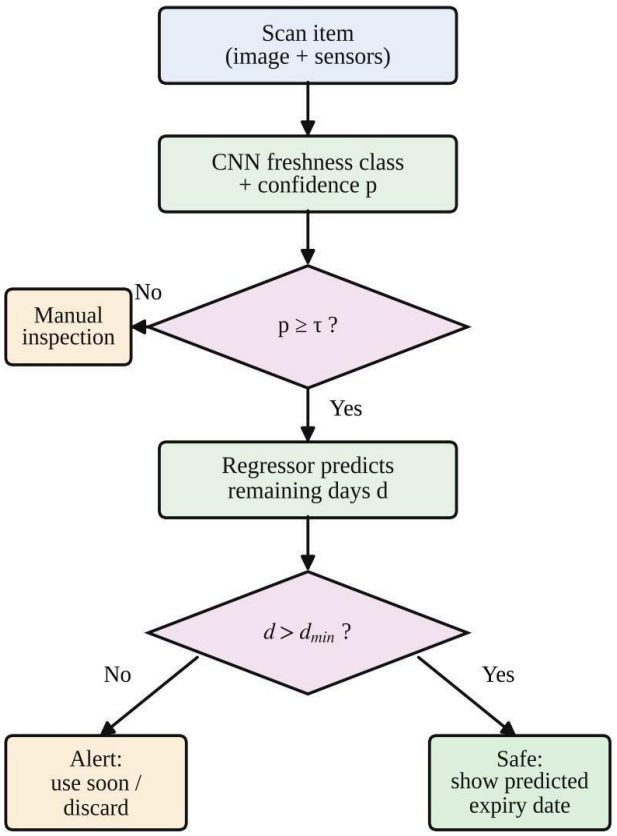


Fig. 5. Confidence-aware decision flow for freshness and expiry alerts.

5. Edge Deployment Architecture

The system is intended to run close to the food (Fig. 6). A camera and the sensor array feed an edge device, a microcontroller or a small single-board computer, which performs sampling, calibration, preprocessing and inference with compressed models. Results are sent over a Wi-Fi or MQTT link to a cloud service that stores data, retrains models and monitors the fleet of devices, and finally to a mobile or web dashboard. Running inference on the device keeps latency low and lets the unit keep working in storage rooms with poor connectivity, in line with the edge approach of [6]. Only compact summaries need to be uploaded, which also reduces bandwidth and the privacy exposure of raw images.

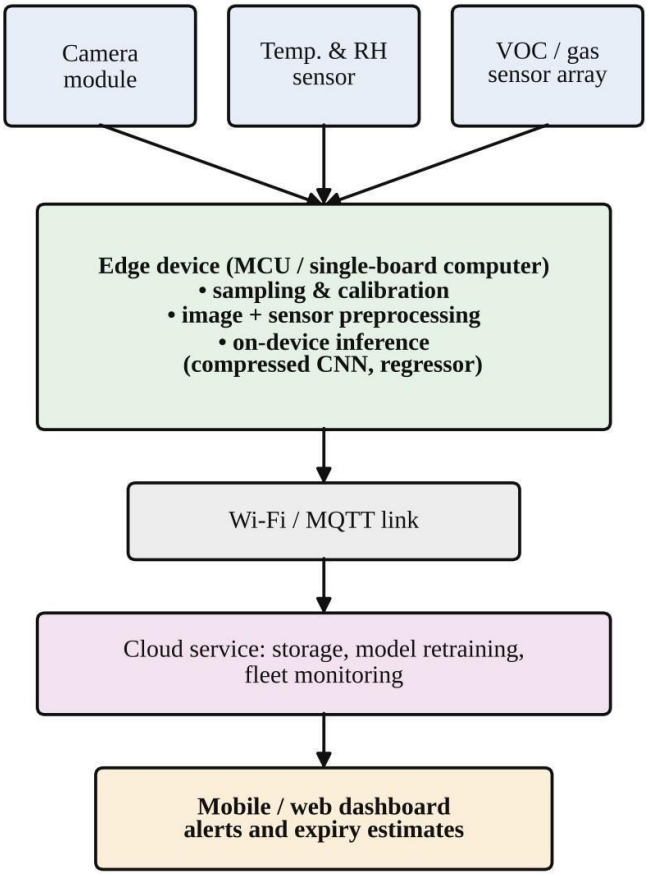


Fig. 6. Edge deployment architecture from sensors to dashboard.

Memory is the first constraint on small devices. Since a weight stored as 32-bit floating point takes four bytes and an 8-bit quantised weight takes one, the parameter counts in Table III translate directly into approximate storage needs (Table VI).

© 2026 IJSRET

MobileNetV2 shrinks from roughly 14 MB to about 3.4 MB after quantisation, while ResNet-50 stays an order of magnitude larger. These figures cover weights only and exclude activations and runtime overhead, but they explain why a compact backbone is the default for the edge unit and why a larger model, if used, would run in the cloud on uploaded images.

Table 6: Approximate Weight Storage (Derived from Parameter Counts)

Backbone	Parameters	float32 (≈)	int8 (≈)
ResNet-50	≈26 M	≈104 MB	≈26 MB
MobileNetV2	≈3.4 M	≈13.6 MB	≈3.4 MB
EfficientNet-B0	≈5.3 M	≈21 MB	≈5.3 MB

VI. EXPERIMENTAL DESIGN

1. Data Collection

Public fresh-versus-rotten fruit image datasets can be used to pre-train the CNN. For expiry prediction, storage experiments are proposed in which several kinds of produce are kept at different controlled temperature and humidity levels, with images and sensor readings logged at fixed intervals until spoilage is confirmed by an expert or a laboratory test. The confirmed spoilage day gives the regression label. Including both cold and warm runs is important so that the models see the temperature effect illustrated in Fig. 1.

2. Preprocessing and Training

Images are normalised and augmented by flipping, rotation and brightness changes to reduce sensitivity to lighting. Sensor signals are denoised with a moving average, baseline-calibrated and min–max scaled. The planned training configuration is listed in Table VII; these are starting values to be tuned on validation data, not reported results.

Sensor noise is reduced with a moving average over a window of w samples, after which each channel is scaled to the unit interval using its minimum and maximum on the training runs:

$$y\text{avg},t = (1/w)\ \Sigma j = 0 \ldots w-1\ yt-j \quad (7)$$

$$y' = (y - y\text{min}) / (y\text{max} - y\text{min}) \quad (8)$$

Using only training-run minima and maxima avoids leaking information from the test runs. Baseline correction subtracts each sensor's reading in clean air recorded at the start of a run, which reduces the effect of drift between sessions.

Table 7: Planned Training Configuration

Component	Planned setting
Input / augmentation	224 × 224 RGB; flip, rotation, brightness
CNN optimiser	Adam; learning rate 10^{-4} for fine-tuning
CNN schedule	Batch 32; up to 30 epochs, early stopping
CNN regularisation	Dropout 0.3; weight decay
Regressors	Random forest [16]; gradient boosting [17]
Hyper-parameter search	Grid / random search on validation runs
Validation	k-fold split by storage run (k = 5)

3. Evaluation Protocol

To avoid data leakage, training, validation and test sets are split by storage run rather than by individual image, since images from the same run are strongly correlated (Fig. 7).

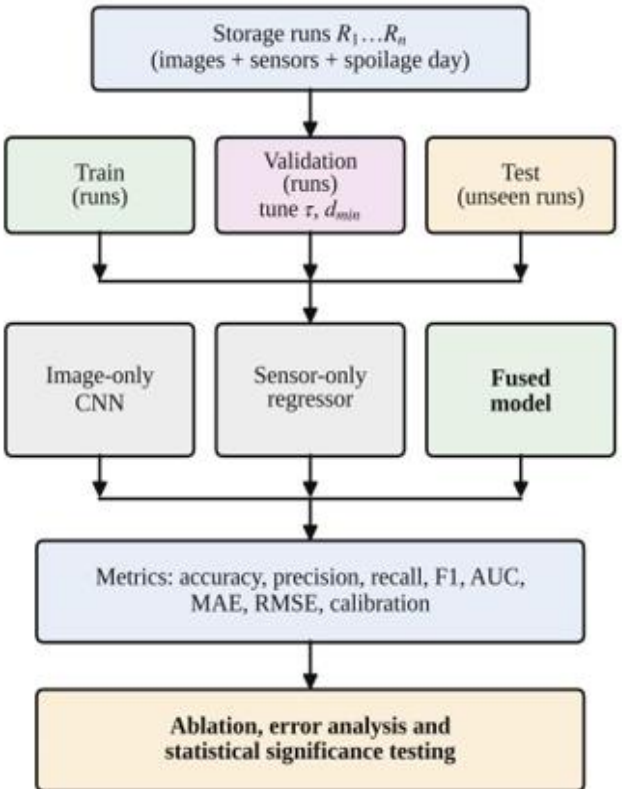


Fig. 7. Planned evaluation protocol with run-wise splits and three model variants.

© 2026 IJSRET

The threshold τ and the minimum shelf life dmin are tuned on the validation runs only. Three variants are compared: an image-only CNN, a sensor-only regressor, and the fused model. This ablation shows what each modality contributes. Error analysis by product type, temperature and lighting condition, together with statistical significance testing across folds, completes the protocol.

Algorithm 1. Threshold selection on validation runs.

- For each candidate τ on a grid from 0 to 1, route every validation item with p < τ to manual inspection.
- On the remaining items, compute the recall on the spoiled class and the share of items sent to inspection.
- Keep only thresholds whose recall meets the target and whose inspection share stays within the available budget.
- Choose the smallest remaining τ, that is, the one that inspects the fewest items.
- Select dmin in the same way by sweeping it against confirmed spoilage days, then report both values on the held-out test runs once.

Raising τ sends more items to inspection and lowers the rate of unsafe automatic decisions, so the choice is a trade-off between inspection effort and risk (Fig. 8). The schematic curves show the direction of the effect only; the actual shapes depend on how well calibrated the CNN is.

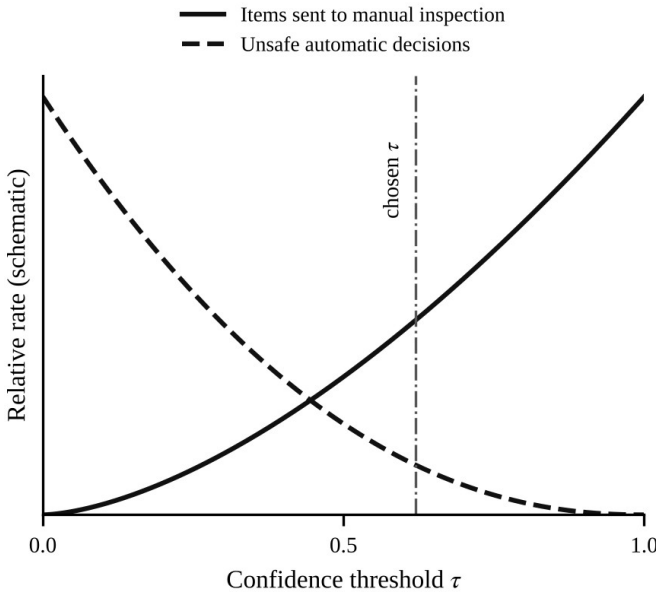


Fig. 8. Schematic trade-off between inspection effort and unsafe automatic decisions as the threshold varies (not measured data).

4. Evaluation Metrics

Classification is assessed with accuracy, precision, recall, F1-score and AUC, where precision, recall and F1 are

$$Precision = TP / (TP + FP) \quad (9)$$

$$Recall = TP / (TP + FN) \quad (10)$$

$$F1 = 2 \cdot Precision \cdot Recall / (Precision + Recall) \quad (11)$$

Recall on the spoiled class is given priority. Expiry prediction is assessed against the observed spoilage day di using the mean absolute error and root mean squared error over n items:

$$MAE = (1/n) \Sigma \mid d\mathrm{i} - d\mathrm{pred,i} \mid \quad (12)$$

$$RMSE = \sqrt{[\ (1/n) \Sigma\ (\ d\mathrm{i} - d\mathrm{pred,i}\ )^2\ ]} \quad (13)$$

Calibration of the confidence p is also reported, because the decision rule relies on it. It is summarised by the expected calibration error over M confidence bins Bm:

$$ECE = \Sigma\mathrm{m}\ (\ \mid B\mathrm{m} \mid / n\ )\ \mid \mathrm{acc}(B\mathrm{m}) - \mathrm{conf}(B\mathrm{m}) \mid \quad (14)$$

A low value means that a stated confidence of, say, 0.9 really corresponds to correct decisions about nine times in ten. No experimental results are claimed in this paper; the protocol above defines the validation planned for the framework.

VII. COMPARATIVE ANALYSIS AND DISCUSSION

Table I shows that the approaches differ in modality, task and deployment, so their reported figures are not directly comparable: they use different products, datasets and metrics. Several observations still follow. Images are non-invasive and cheap to capture and let CNNs avoid manual feature engineering [4], [5], but they mainly capture surface appearance. Gas sensing responds to volatile compounds released during spoilage and may reveal change before it is visible [3], [2], [6]. Regression models turn a freshness state into a number of days, the quantity that retailers and consumers actually need [2]. Edge deployment keeps cost and latency low and suits places with poor connectivity [6].

Because vision and sensing capture complementary signals, a fused system is expected to be more robust than either alone; this expectation remains to be tested by the ablation in Section VI-C. A second expected benefit is operational. Item-level estimates could let a retailer rotate stock by actual condition

© 2026 IJSRET

instead of by printed date, and could let a household decide when to cook or discard. By replacing static labels with predictions that respond to storage conditions, such a system can reduce unnecessary disposal of edible food and also catch unsafe food that a printed date would miss.

There are also governance questions. A system that influences what is thrown away or sold carries responsibility if it errs, so its outputs should be presented as decision support with a visible confidence level, not as a guarantee. Any commercial use would also need to be reconciled with food-labelling rules, which this paper does not address.

Table VIII sketches three deployment scenarios and the benefits expected from each. These are expectations drawn from the reviewed work, not measured outcomes. In a household, an alert that an item should be eaten soon helps avoid throwing away good food. In a retail cold room, ordering stock by observed condition rather than by delivery date could reduce both waste and the sale of items close to spoilage. In a warehouse or cold chain, tracking temperature and gas signatures can flag batches that have been stored warm, so that they are inspected or sold first.

Table 8: Deployment Scenarios and Expected Benefits

Scenario	Inputs	Decision	Expected benefit
Household fridge	Camera, gas, temperature	Use-soon alert	Less discard of edible food
Retail cold room	Camera, gas, temperature / RH	Rotate stock by condition	Fewer wasted or unsafe items
Warehouse / cold chain	Temperature / RH, gas	Flag batches stored warm	Earlier detection of spoilage risk

Reducing avoidable food waste also saves the water, energy and labour that went into producing and transporting the food, which is why the issue is treated as a sustainability concern [1]. A condition-based approach is only one lever among many, since better cold-chain practice and better packaging address the same problem at its source. Item-level prediction is best seen as a complement that makes the remaining waste visible and actionable.

Challenges and Threats to Validity

Table IX lists the main risks and possible mitigations. Data limitations are the most important: image datasets are often captured against controlled backgrounds, and sensor datasets are small and product-specific, so models may generalise poorly. Low-cost gas sensors drift and respond to many compounds, so periodic calibration and temperature compensation are required. A false "fresh" prediction on spoiled food can cause harm, so conservative thresholds, confidence-aware routing and human inspection should be standard. Packaging can hide products from cameras, and the system must be affordable for small vendors and households.

Table 9: Main Risks and Mitigations

Risk	Effect	Mitigation
Dataset bias	Poor generalisation to new products or lighting	Multi-product data; augmentation; domain testing
Sensor drift	Predictions degrade over weeks	Periodic calibration; baseline correction
False "fresh"	Unsafe food judged safe	Recall-oriented τ; manual inspection route
Occlusion	Packaging hides produce from camera	Sensor-only fallback mode
Cost	Too expensive for small vendors	Low-cost MCU; compressed models
Trust	Users distrust black-box output	Grad-CAM heat-maps [18]; confidence display

Threats to the validity of this study itself should also be stated. The review covers a limited number of sources found through web search and may miss relevant work. The reported accuracies come from different products, datasets and metrics and cannot be ranked against one another. Finally, the proposed framework has not been implemented, so its benefits remain hypotheses until the planned experiments are completed.

Future Directions

Several directions follow from this work. The first is to build and release a multi-product dataset that pairs images and sensor logs with confirmed spoilage days under realistic supply-chain conditions. The second is to implement the framework on low-cost hardware and test it in a cold room and a retail setting. The third is explainability: heat-maps from methods such as Grad-CAM [18] can show which regions of an image drove afreshness decision and help users trust it. The fourth is integration with cold-chain monitoring and logistics systems [12], so that predicted remaining shelf life can guide routing and stock rotation. Finally, model compression and quantisation can bring the CNN to very small devices [9], and spectral imaging [11] could be added where its cost is justified. Fig. 9 gives an indicative six-month plan for carrying the framework from design to evaluation. Hardware assembly and

© 2026 IJSRET

calibration come first, because the quality of the sensor data limits everything downstream. Storage-run data collection then overlaps with CNN training, so that early runs can be used to debug the pipeline. Fusion, edge deployment and evaluation close the plan. The durations are estimates for a small team and would change with access to storage facilities and laboratory confirmation of spoilage.

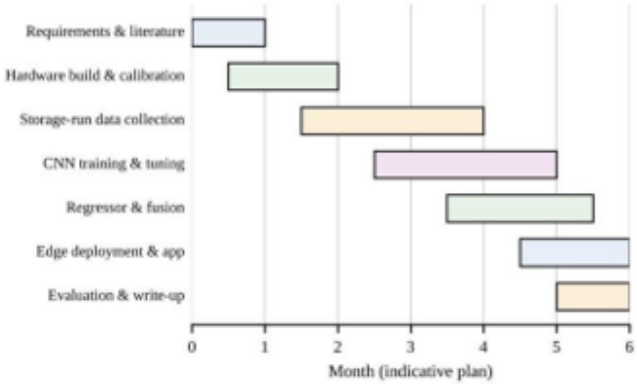


Fig. 9. Indicative six-month roadmap for implementing and evaluating the framework.

VIII. CONCLUSION

This paper reviewed AI-based freshness detection and expiry prediction and proposed a vision–sensor fusion framework with confidence-aware decision logic and an edge-deployment architecture. Published work shows that CNNs can assess freshness from images and that low-cost gas sensors with ML models can predict spoilage, with reported accuracies of 91.9% to 97.05% in the sensor-based studies [3], [6]. Combining the two modalities in an edge-deployable pipeline can provide item-specific expiry estimates that reduce food waste and improve food safety, provided that unsafe errors are controlled through calibrated confidence and human review. The next step is to validate the framework on a multi-product dataset under realistic storage conditions.

REFERENCES

1. J. Gustavsson, C. Cederberg, U. Sonesson, R. van Otterdijk, and A. Meybeck, Global Food Losses and Food Waste: Extent, Causes and Prevention. Rome, Italy: Food and Agriculture Organization of the United Nations, 2011.
2. P. Wunderlich et al., “Enhancing shelf life prediction of fresh pizza with regression models and low cost sensors,” Foods, vol. 12, no. 6, Art. no. 1347, Mar. 2023, doi: 10.3390/foods12061347.
3. K. Nair, B. Sekhani, K. Shah, and S. Karamchandani, “Expiry prediction and reducing food wastage using IoT and ML,” Int. J. Elect. Comput. Eng. Syst., vol. 12, no. 3, pp. 155–162, 2021, doi: 10.32985/ijeces.12.3.4.
4. C. Wang, S. Liu, Y. Wang, J. Xiong, et al., “Application of convolutional neural network-based detection methods in fresh fruit production: A comprehensive review,” Front. Plant Sci., vol. 13, Art. no. 868745, May 2022, doi: 10.3389/fpls.2022.868745.
5. M. Marcuzzo et al., “Fruit ripeness classification: A survey,” arXiv:2212.14441, 2022.
6. “Machine learning-based shelf life estimator for dates using a multichannel gas sensor: Enhancing food security,” Sensors, vol. 25, no. 13, Art. no. 4063, Jun. 2025, doi: 10.3390/s25134063.
7. S. J. Pan and Q. Yang, “A survey on transfer learning,” IEEE Trans. Knowl. Data Eng., vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
8. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2009, pp. 248–255.
9. “From tiny machine learning to tiny deep learning: A survey,” arXiv:2506.18927, 2025.
10. N. Gautam and N. Chaurasia, “Forecasting of fruits stock life using CNN-based deep learning techniques: A comprehensive study,” in CEUR Workshop Proc., vol. 3283, 2022, paper 63.
11. “Hyperspectral imaging and deep learning for quality and safety inspection of fruits and vegetables: A review,” 2025. [Online]. Available: https://acs.figshare.com/articles/journal_contribution/28806553
12. “IoT-based fresh produce supply chain under uncertainty: An adaptive optimization framework,” arXiv:2511.05920, 2025.
13. K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 770–778.
14. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted residuals and linear bottlenecks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018, pp. 4510–4520.
15. M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” in Proc. 36th Int. Conf. Mach. Learn. (ICML), PMLR vol. 97, 2019, pp. 6105–6114.
16. L. Breiman, “Random forests,” Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.

© 2026 IJSRET

17. T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2016, pp. 785–794.
18. R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-CAM: Visual explanations from deep networks via gradient-based localization,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), 2017, pp. 618–626.

© 2026 IJSRET