IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, November 2026
Impact Factor: 8.2

FreshnessAI: AI-Based Freshness and Shelf-Life Prediction for Cut Fruits

Angel Rose T S[1], Devika A S[2], Jesna M Thomas[3], Sneha Johnson[4], Anjitha P[5]
Student, Department of Computer Science and Engineering[1,2,3]
Assistant Professor, Department of Computer Science and Engineering[4,5]
Thejus Engineering College, Thrissur, Kerala, India

Abstract : *Cut fruits are minimally processed products that deteriorate quickly after cutting, and shops and food businesses usually judge their freshness by eye. Such inspection is subjective and gives no reliable estimate of how long a piece of cut fruit can still be sold or served safely. This paper presents FreshnessAI, a proposed web and mobile system that applies computer vision and deep learning to assess the freshness of cut fruits from images and to provide shelf-life (expiry) guidance. The system is built around a Python Django backend with two clients: a web application for administrators, fruit shops and restaurants, and a Flutter-based Android application for customers. An AI inference engine built with PyTorch and YOLO pre-processes an uploaded image, detects the fruit type and location, assigns a freshness status (fresh or spoiled) with a confidence score, and produces a shelf-life estimate with advice. Results and images are stored in a MySQL database and media storage, and the YOLO model is trained offline through a pipeline of dataset collection, annotation, augmentation, splitting and GPU-based training. We survey twenty research papers, identify the gaps they leave for cut fruits, and present the requirements, architecture and data flow design. Dataset preparation, training and evaluation are planned for the next phase, and no performance figures are claimed in this paper..*

Keywords: Cut fruits, freshness classification, shelf-life prediction, YOLO, deep learning, computer vision, Django, Flutter.

I. INTRODUCTION

Fruits form an important part of a healthy diet, and a growing share of them is sold ready to eat as sliced fruit boxes, fruit salads and cut fruit served in juice bars, restaurants and catering services. Once a fruit is peeled and cut, its protective skin is removed and the exposed flesh changes quickly in colour, texture and smell. Cutting damages the cells of the fruit, which accelerates enzymatic browning, water loss, softening and the growth of micro-organisms, so the useful life of cut fruit is much shorter than that of the whole fruit.

Fruit shops, juice bars and restaurants must therefore decide repeatedly whether a batch of cut fruit is still fresh enough to be sold or served. A wrong decision has a cost in both directions: discarding good fruit wastes food and money, while serving spoiled fruit harms customers and their trust. In practice the decision is taken by looking at the fruit. Inspection based on human judgement is subjective, time-consuming and costly, and has been described as a limitation in studies of fruit freshness assessment [18], [20]. Two inspectors can classify the same slice differently, the same inspector may judge differently under different lighting, and a visual impression says little about how much useful life remains.

Computer vision and deep learning make it possible to automate such visual judgements. Convolutional neural networks (CNNs) learn visual features directly from labelled images and have been applied to fruit freshness and ripeness classification [2], [5], [7], [20]. Object detectors such as YOLO (You Only Look Once) predict both the class and the location of objects in one pass, which makes fast detection practical [3], [10]. Estimating the remaining shelf life is a harder task than classifying freshness: it needs training data in which the time to spoilage is known, and it is

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, November 2026
Impact Factor: 8.2

sensitive to storage conditions. Among the studies we reviewed, only RipeTrack estimates remaining lifetime, and it relies on near-infrared data and targets whole climacteric fruits [4].

This paper presents FreshnessAI, a proposed web and mobile system that brings these ideas together for cut fruits. The main contributions of this Phase I work are:

- A survey of twenty recent studies on fruit freshness, ripeness, detection and shelf-life prediction, and the identification of five research gaps relevant to cut fruits.
- A design that treats fruit detection, freshness classification and shelf-life prediction as three distinct tasks within one AI inference workflow.
- A multi-user system architecture, with a Django backend, a web application for business users and a Flutter-based Android application for customers, together with an offline model training pipeline and Level 0 and Level 1 data flow diagrams.

The remainder of the paper is organized as follows. Section II reviews the literature, Section III describes the system design, and Section IV concludes with future work.

II. LITERATURE SURVEY

We reviewed twenty studies published between 2025 and 2026. They are grouped below by the problem they address and summarized in Table I.

TABLE I: Summary of Reviewed Literature by Theme

Theme	Refs.	Typical limitation
Freshness / ripeness classification	[2], [5], [7], [9], [12], [18], [20]	Controlled data, few fruits, labels only, no cut fruit, no shelf life
Fruit detection	[3], [10], [17]	Whole fruit or orchard; few fruit types; complex backgrounds
Special sensors / imaging	[4], [11], [13], [15], [19]	Extra hardware and cost; limited real-world validation
Defects and sorting	[1], [8]	No cut-fruit freshness or shelf life
Limited data / robustness	[6], [14], [16]	Needs more categories, validation data and robustness

Freshness and Ripeness Classification

Hashemi and Amiri combine machine learning, image processing and CNNs to predict fruit quality, but need controlled backgrounds and cover limited fruit diversity [2]. PapayaFreshNet joins ResNet50, a Transformer and squeeze-and-excitation blocks for non-destructive papaya classification, yet is limited to papaya and Good/Bad labels [5]. CNN models for ripeness classification [7] and the RipeNet model [12] automate manual inspection but do not analyse cut fruit or predict shelf life. A machine-vision approach to freshness and ripeness [18] still needs more diverse datasets and better real-world freshness assessment. Fischer-Brandies et al. use deep learning with Gaussian filtering to separate fresh from rotten fruit [9], and Ghosh and Singh propose a hybrid CNN-LSTM freshness classifier, which is limited by fruit diversity, controlled data and computational cost [20].

Object Detection of Fruit

Ait Issad et al. evaluate YOLOv8, YOLOv9 and YOLOv10 for fruit maturity detection, which locates the fruit and assigns a maturity stage in one pass, but few fruit types and complex backgrounds remain a weakness [3]. Kamat et al. compare YOLO and SSD for multi-class ripeness detection on whole fruits [10]. Dutta et al. integrate apple disease diagnosis, freshness assessment and fruit detection in a low-cost UAV pipeline for orchards [17].

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, November 2026
Impact Factor: 8.2

Specialised Sensors and Imaging

RipeTrack estimates ripeness and remaining lifetime on a smartphone using RGB images and near-infrared (NIR) data, but mainly targets climacteric fruits [4]. Other work relies on gas sensors for sorting baskets [11], hyperspectral imaging for apple firmness and Brix [13], and thermal imaging for banana sorting [15]. A review of AI-augmented non-destructive technologies notes that high sensor cost, limited robustness in real conditions and a lack of standardisation restrict large-scale deployment [19].

Defect Detection and Sorting

SmartDate automates the sorting and quality control of date fruits [1], and ANN-SVM-IP detects and classifies external apple defects [8]. Neither addresses cut-fruit freshness or shelf life.

Limited Data and Robustness

Jain et al. use privacy-preserving ordinal meta-learning with vision-language models for fine-grained fruit quality when labels are scarce [6], and Kim and Sun apply balanced multi-source transfer learning when little target-domain data is available [14]. He et al. extract fine-grained food-image features with iterative clustering and attention, noting that robustness to lighting and background must still improve [16].

Research Gaps Identified

The review reveals the following gaps.

1. No cut-fruit focus. Almost all work targets whole fruits. Cut fruit spoils within hours and its visual cues differ from those of intact fruit; studies that address freshness classes [5], [9], [20] do not treat cut surfaces.

2. Classification instead of prediction. Most studies output ripeness stages or fresh/rotten labels [3], [5], [7], [9], [10], [12], [15], [20]. Only [4] estimates remaining lifetime, and it needs NIR data.

3. Dependence on special hardware. Thermal cameras [15], hyperspectral imagers [13], gas sensors [11], NIR [4] and UAVs [17] raise cost and limit adoption [19].

4. Limited robustness and diversity. Controlled backgrounds, few fruit types and weak performance under varying lighting, viewpoint and clutter are repeatedly reported [2], [3], [13], [15], [16], [20].

5. Scarce labelled data. Fine-grained freshness labels are hard to obtain, which motivates few-shot and transfer learning [6], [14].

FreshnessAI addresses these gaps by targeting cut fruits with an accessible image-only workflow that needs no extra sensors.

III. SYSTEM DESIGN

Overview and Users

FreshnessAI is a web and mobile platform that assesses the freshness of cut fruit from an image and gives expiry (shelf-life) guidance. It has four types of users: the Admin, who verifies shops and restaurants, manages fruit categories, generates reports and sends notifications; the User (customer), who scans cut-fruit images in the mobile application, searches for and buys fruit, pays, tracks orders and gives feedback; the Shop (fruit vendor), who uploads images, manages inventory and handles orders; and the Restaurant, who uploads images, checks expiry information and receives notifications. Administrators, shops and restaurants use a web application, while customers use a Flutter-based Android application. Both clients use the same Django backend and AI inference engine, and no additional hardware is needed on the client side.

System Architecture

Fig. 1 shows the architecture, which has five parts: the presentation layer, the application layer, the AI inference engine, the data layer and the offline model training pipeline.

The presentation layer contains the four user types and two clients: a browser client built with HTML5, CSS3, Bootstrap and JavaScript, and a Flutter mobile client. Both communicate with the application layer over HTTP/HTTPS.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, November 2026
Impact Factor: 8.2

The application layer is a Django (Python 3.10+) application that follows the Model–View–Template pattern. It provides URL routing, authentication with role-based access and template rendering, and hosts four role modules (Admin, User, Shop, Restaurant) and four shared modules (Order and Payment, Feedback and Complaints, Notifications, Report Generation). The data layer has a MySQL database for users, shops, restaurants, fruits, detections, carts, purchases, order items, payments, feedback and notifications, and a media store for uploaded images; the application reads and writes it through the Django ORM.

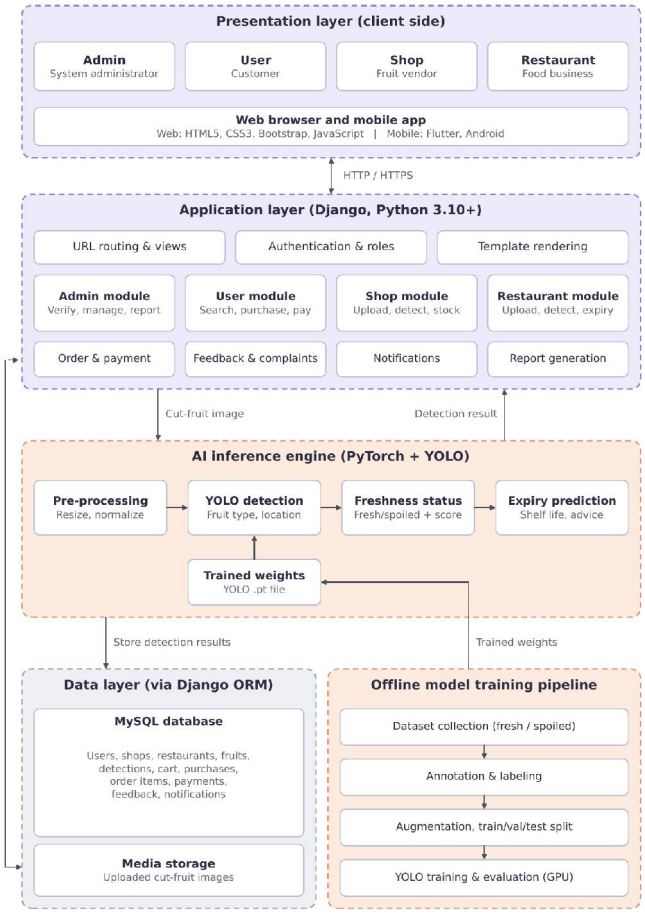


Fig. 1. System architecture of FreshnessAI.

AI Inference Engine

The application layer passes each uploaded cut-fruit image to the inference engine (PyTorch and YOLO), which works in four stages: (1) *pre-processing* resizes and normalizes the image; (2) *YOLO detection* returns the fruit type and location using trained weights loaded from a .pt file; (3) *freshness status* gives a fresh or spoiled label with a score; and (4) *expiry prediction* produces the shelf life and advice. The result returns to the application layer and is stored in the data layer.

The three tasks behind these stages are distinct and are treated separately (Table II). Detection can be learned from images with bounding-box labels. Freshness classification can be learned from freshness-specific class labels or by a separate classifier; this choice will be made in Phase II after experiments. Shelf-life prediction cannot be obtained from detection alone: it needs observations in which the time to spoilage is known, a validated prediction method, and

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, November 2026
Impact Factor: 8.2

information such as fruit type and storage condition. FreshnessAI therefore presents shelf life as an estimate with advice, and will report it only after it has been validated.

TABLE II: Detection, Freshness Classification and Shelf-Life Prediction

Task	Output	Main requirement
Object detection	Fruit type and bounding box	Images with bounding-box labels
Freshness classification	Fresh/spoiled status and score	Images labelled fresh or spoiled
Shelf-life prediction	Estimated remaining time and advice	Time-to-spoilage observations and a validated method

Offline Model Training Pipeline

The model is not trained inside the web application. The offline pipeline has four steps: dataset collection (fresh and spoiled cut-fruit images), annotation and labelling, augmentation with a train/validation/test split, and YOLO training and evaluation on a GPU. Augmentation enlarges a small dataset by transforming images, and the split allows evaluation on unseen images. To avoid optimistic evaluation, all images of the same fruit piece are to be kept in the same split. The resulting trained weights are supplied to the inference engine, so the model can be improved and replaced without changing the web application.

Data Flow

Fig. 2 shows the Level 0 (context) diagram, in which FreshnessAI is a single process exchanging data with four external entities. The Admin sends login details and verification actions and receives the login status and reports; the User sends login and registration details, orders, payments and images and receives the login status, order status and results; the Shop sends login and registration details, stock details and images and receives the login status, results and stock information; and the Restaurant sends login and registration details and fruit images and receives the login status, expiry information and alerts. Fig. 3 shows the Level 1 diagram, which decomposes this context diagram into five processes and five data stores. Process 1.0 handles login and verification against the accounts store D1. Processes 2.0 (User), 3.0 (Shop) and 5.0 (Restaurant) serve their own entities, and the User, Shop and Restaurant processes all pass images to the shared AI detection process 4.0. Process 4.0 loads the trained weights from D5, applies pre-processing, detection, freshness status and expiry prediction, stores detections, results and images in D3, sends expiry alerts to the notifications store D4, and returns the result to the process that sent the image. D2 holds inventory and orders.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, November 2026
Impact Factor: 8.2

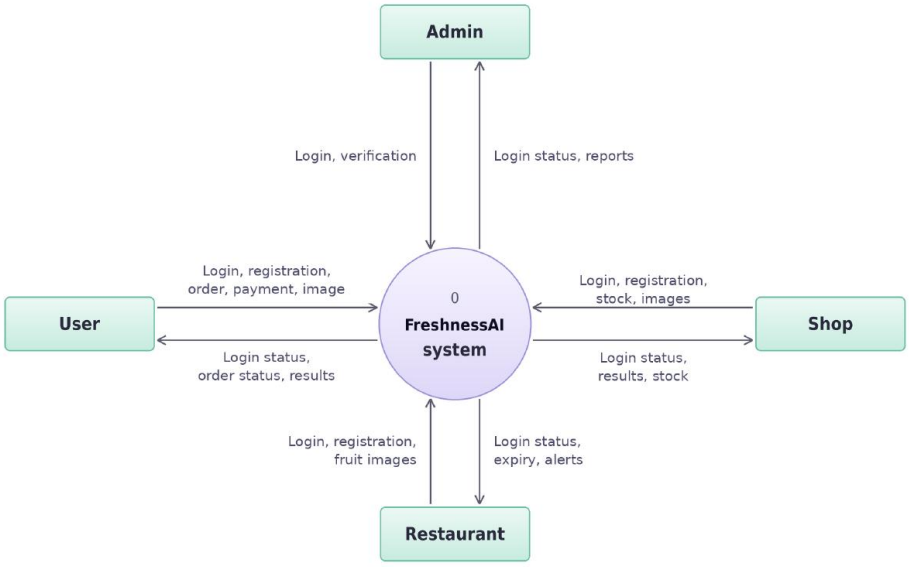


Fig. 2. Level 0 data flow diagram of FreshnessAI.

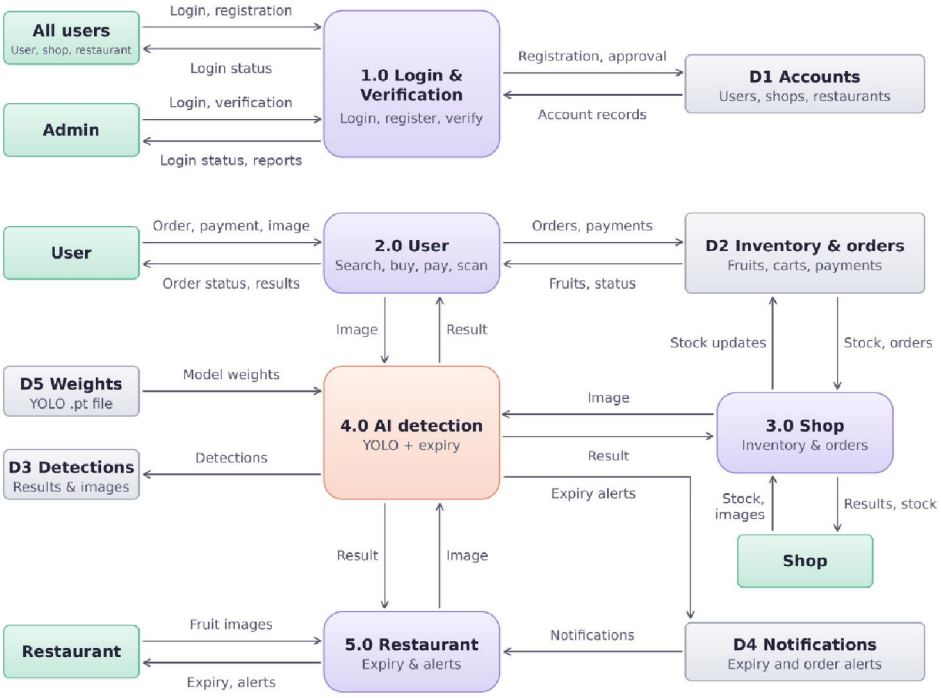


Fig. 3. Level 1 data flow diagram of FreshnessAI.

Requirements

The proposed requirements follow from the architecture. Training a YOLO model is the most demanding task and is expected to need an NVIDIA CUDA-capable GPU (at least 6 GB VRAM recommended), 16 GB or more of RAM and a 512 GB SSD, whereas running a small trained model on one image can be done on a CPU. The software stack

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 2, November 2026 Impact Factor: 8.2

comprises Python 3.10+, Django, PyTorch, YOLO, MySQL, HTML5/CSS3/Bootstrap/JavaScript for the web front end, Flutter (Dart) for the Android application, and Visual Studio Code with Git and GitHub for development. Exact figures are to be confirmed once the model and dataset size are final.

Status of the Work

In Phase I the problem has been studied, the requirements defined, and the architecture and data flow diagrams designed. The components described in this section are planned components of the proposed system. Dataset collection, model training, integration and evaluation belong to the next phase, and this paper makes no claim about trained models or measured performance.

IV. CONCLUSION AND FUTURE WORK

This paper presented the Phase I design of FreshnessAI, a web and mobile system for assessing the freshness of cut fruits and giving shelf-life guidance. A survey of twenty studies showed that freshness classification of whole fruit is well studied, that YOLO-family detectors suit fruit localization, and that shelf-life prediction from images is possible only with time-based datasets and careful validation. Very few studies address cut fruit, and none of those reviewed combines detection, freshness status and shelf-life guidance in a multi-user web and mobile application. The proposed system uses a Django backend, a web application and a Flutter-based Android application, with a PyTorch and YOLO inference engine, a MySQL database with media storage, and an offline training pipeline. At the end of Phase I the work consists of analysis and design only.

Future work includes the following.

Dataset improvement: collect and annotate more fresh and spoiled cut-fruit images under different lighting, backgrounds, cutting styles and storage conditions.

More fruit types: start with a small set of commonly sold cut fruits and extend to more fruits and mixed fruit salads.

Finer spoilage classes: move from fresh/spoiled to intermediate freshness levels, as in earlier grading studies [3], [15].

Shelf-life estimation: record images at regular intervals until spoilage, with storage temperature and time, to obtain labelled data for a validated shelf-life model, and consider explanation tools.

Model validation: evaluate the trained model with detection metrics such as mAP and with classification metrics on a held-out test set, keeping all images of one fruit piece in the same split.

Model choice and deployment: compare YOLO model sizes for accuracy against response time [3], [10], deploy the system, and test it with real shops, restaurants and consumers.

Business features: extend inventory, notifications and reports so that wastage trends can be seen and stock planned.

V. ACKNOWLEDGMENT

The authors express their sincere gratitude to Dr. Vince Paul, Principal, and Ms. Vinitha A V, Head of the Department of Computer Science and Engineering, Thejus Engineering College, Thrissur, for their encouragement and support throughout this work. The authors are thankful to their project guide, Ms. Sneha Johnson, for her expert supervision, encouragement and constructive criticism, and to the project coordinator, Ms. Anjitha P, for the valuable coordination and support provided during the project. The authors also acknowledge the help provided by the members of the college library and the institutional staff, and thank their classmates for their support.

REFERENCES

1. K. Eskaf, “SmartDate: AI-driven precision sorting and quality control in date fruits,” 2025. https://doi.org/10.48550/arXiv.2508.00921
2. A R. Hashemi and S. Amiri, “Predicting fruit quality with a hybrid machine learning and image processing approach,” 2026, arXiv:2606.26165.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, November 2026
Impact Factor: 8.2

3. H. Ait Issad, S. Lazib, S. Oubabas, K. Chemoun, R. Aoudjit, and J. J. P. C. Rodrigues, "Accurate fruit maturity detection using advanced object detection techniques," in *Proc. 5th Int. Conf. Electr., Comput. Energy Technol. (ICECET)*, Paris, France, Jul. 2025, doi: 10.1109/ICECET63943.2025.11472300.
4. M. S. Waseem, N. Sharma, and M. Hefeeda, "RipeTrack: Assessing fruit ripeness and remaining lifetime using smartphones," *IEEE Trans. Mobile Comput.*, vol. 25, no. 1, pp. 830–845, Jan. 2026, doi: 10.1109/TMC.2025.3599917.
5. A Sar, T. Choudhury, T. Sarkar, and K. Kotecha, "PapayaFreshNet: A hybrid deep learning framework for non-destructive freshness classification of papayas using convolutional and transformer networks," *Discover Food*, vol. 5, Art. no. 97, 2025, doi: 10.1007/s44187-025-00368-9.
6. R. Jain, M. Patwardhan, A. Mishra, P. Deshpande, and B. Rai, "Privacy preserving ordinal-meta learning with VLMs for fine-grained fruit quality prediction," Nov. 2025, arXiv:2511.01449v1 [cs.CV].
7. Bh. Maheedhar and S. Lalitha, "Ripeness revolution: Harnessing CNN models for precise fruit classification," in *Proc. 3rd Int. Conf. Data Sci. Inf. Syst. (ICDSIS)*, 2025, doi: 10.1109/ICDSIS65355.2025.11070765.
8. N. M. Hussain Hassan, M. H. M. Mahmoud, M. A. Ismeil, M. M. Mabrook, A. A. Donkol, and A. M. Mabrouk, "ANN-SVM-IP: An innovative method for rapidly and efficiently detecting and classifying of external defects of apple fruits," *IEEE Access*, vol. 13, pp. 123487–123514, 2025, doi: 10.1109/ACCESS.2025.3586967.
9. L. Fischer-Brandies, L. Müller, J. J. Riegger, and R. Buettner, "Fresh or rotten? Enhancing rotten fruit detection with deep learning and Gaussian filtering," *IEEE Access*, vol. 13, pp. 31857–31869, 2025, doi: 10.1109/ACCESS.2025.3542612.
10. P. Kamat, S. Gite, H. Chandekar, L. Dlima, and B. Pradhan, "Multi-class fruit ripeness detection using YOLO and SSD object detection models," *Discover Appl. Sci.*, vol. 7, Art. no. 931, 2025, doi: 10.1007/s42452-025-07617-7.
11. M. R. Anirudh, "Smart fruit sorting basket: Vision-gas sensor fusion for ripeness classification and automated sorting," in *Proc. Int. Conf. Responsible, Generative and Explainable AI (ResGenXAI)*, 2025, doi: 10.1109/RESGENXAI64788.2025.11344032.
12. K. Bhavana, R. Sumathi, K. Vanaja, S. Manjula, T. Srinivasulu, and S. Divya, "Fruit ripeness detection using RipeNet," in *Proc. Int. Conf. Electron. Renew. Syst. (ICEARS)*, 2025, pp. 1598–1604, doi: 10.1109/ICEARS64219.2025.10940159.
13. H. Tang, Y. Liu, I. Boukhennoufa, W. Yu, X. Xu, and X. Zhai, "Toward accurate prediction of apple firmness and Brix across countries, seasons and cultivars with hyperspectral imaging," *Discover Food*, vol. 6, Art. no. 335, Jun. 2026, doi: 10.1007/s44187-026-01072-y.
14. Y. Kim and E. J. Sun, "Balanced multi-source transfer learning for fruit quality prediction with limited data," *IEEE Access*, vol. 14, pp. 87992–88008, 2026, doi: 10.1109/ACCESS.2026.3697751.
15. T. Y. Melesse, "Intelligent postharvest sorting of bananas using thermal imaging and deep neural network models," *Food Bioprocess Technol.*, vol. 19, Art. no. 80, 2025, doi: 10.1007/s11947-025-04173-1.
16. W. He, M. Liu, and Q. Liang, "Capturing fine-grained food image features through iterative clustering and attention mechanisms," *IEEE Access*, vol. 13, pp. 69888–69897, 2025, doi: 10.1109/ACCESS.2025.3561465.
17. S. Dutta, S. Banerjee, S. Mahata, A. Sen, and S. Datta, "A low-cost UAV deep learning pipeline for integrated apple disease diagnosis, freshness assessment, and fruit detection," 2026, arXiv:2512.22990v2 [cs.CV].
18. N. J. N. Kumar and P. M. Jacob, "Fruit freshness and ripeness detection using machine vision," in *Proc. 1st IEEE Int. Conf. Recent Trends Comput. Smart Mobility (RCSM)*, 2025, doi: 10.1109/RCSM67767.2025.11506784.
19. Y. Fu and D.-W. Sun, "Advancing fruit quality monitoring with artificial intelligence-augmented non-destructive technologies," *Food Eng. Rev.*, vol. 18, Art. no. 9, 2026, doi: 10.1007/s12393-026-09437-w.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, November 2026
Impact Factor: 8.2

20. D. Ghosh and J. P. Singh, "A hybrid CNN-LSTM model for accurate fruit freshness classification using deep learning," *Discover Artif. Intell.*, vol. 6, Art. no. 134, 2026, doi: 10.1007/s44163-025-00750-7.