© DEC 2025 | IRE Journals | Volume 9 Issue 6 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I6-1712938

High-Accuracy Real-Time Defect Classification for Coffee Beans Using Deep Learning

ARCHANA N[1], MAHALAKSMI[2], C G YASHAS RAJ[3], PRAJWAL S B[4]
[1, 2, 3, 4]*Dept. Computer Science and Engineering (IOT cybersecurity including blockchain)), AIET Moodbidri, India*

Abstract*— Consistent quality inspection of coffee beans remains a critical requirement in the agricultural supply chain, as manual sorting is slow, inconsistent, and dependent on worker experience. With recent advancements in deep learning, real-time object recognition models provide an opportunity to automate quality assessment with high accuracy and minimal intervention. This study presents a YOLOv8-based computer vision system designed to detect and classify defective coffee beans, including broken, discolored, insect-damaged, and mold-affected samples. The proposed model is trained on an annotated dataset of roasted and unroasted beans under varying illumination and background conditions. The system demonstrates strong generalization capability, outperforming traditional CNN classification systems in speed and reliability, achieving up to 97.8% detection accuracy at 60 FPS on GPU inference. The study concludes that YOLOv8 can significantly improve the speed and precision of coffee quality analysis, enabling fully automated grading lines for industrial deployment.[2]

***Keywords*— Computer Vision, Coffee Quality Inspection, YOLOv8, Deep Learning, Defect Detection, Industrial Automation.**

I. INTRODUCTION

Coffee remains one of the most commercially important agricultural products in the globe and the value of its market is firmly linked to its cleanliness, intactness and visual appeal of the beans. The international grading standards lay much stress on the lack of flaws like cracks on the surface, discolouration, mould development, insect corrosion, and damages due to unfavourable or overfermentation.

These tests have been conducted over decades, despite the fact that they have been conducted manually by skilled individuals through their keen eye and the experience they have acquired over the years to determine quality.

Although this conventional approach is suitable with small batches, it becomes more unreliable and expensive when it is applied to industry. Working a long shift incurs fatigue, all people do not have the same individual judgment and the lighting situation cannot always be maintained homogenous and the speed of human inspection is a human factor. Such factors usually lead to inconsistency in the grading and delays within the processing chain.

With the further development of automated systems, machine vision and deep learning have become a potent substitute that can overcome these drawbacks. Convolutional neural networks have demonstrated exceptional expertise in the study of the textures, contours, and smaller visual features on the agricultural commodities. Nevertheless, most of the early models did not target specific areas where defects appeared, and they classified the former with yes-or-no.

Modern object detectors (particularly single-shot detectors) have changed this situation because localization and classification can be performed in a single inference step. This enhances the rate of detection, decreases the amount of computation and renders real-time processing to be more practical to industrial lines.

The YOLO line of detectors has continuously set the limits of both speed and accuracy. YOLOv10 is the latest development in the progression series, which has sophisticated extracting features, efficient architecture, and improved detection accuracy. Such enhancements have allowed it to be used in demanding quality control processes where quick and accurate detection of defects is needed.

II. LITERATURE REVIEW

Early attempts to automate the grading of coffee beans were grounded in traditional image-processing techniques, relying on manually crafted descriptors such as color-based histograms, structural measurements, and texture-derived statistics to flag imperfections. Although these early systems performed acceptably under carefully controlled lighting, their reliability dropped sharply when exposed to shifts in illumination, variations in camera configuration, or background noise. Their dependence on handcrafted features also limited adaptability, making it difficult for such systems to generalize beyond fixed experimental conditions [1][2].

As deep learning techniques matured, the field transitioned toward Convolutional Neural Networks (CNNs), which learn layered representations directly from raw imagery rather than relying on predefined visual cues. Numerous studies demonstrated that CNN classifiers substantially surpassed traditional feature-engineered methods in identifying roasting inconsistencies, insect-related damage, and a wide range of surface defects in coffee beans [3] [4]. However, most of these CNN-driven models produced only whole-image predictions and lacked the ability to pinpoint the exact location of defects. This absence of spatial localization limited their practicality for industrial settings where precise per-bean defect positioning is essential for automated sorting.

The need for spatially aware, real-time inspection led researchers to adopt object-detection architectures capable of performing both classification and localization simultaneously. Two-stage detectors like Faster R-CNN delivered high localization accuracy, yet they struggled to meet the speed requirements for fast-moving conveyor- based inspection lines [5]. Single-stage detectors from the YOLO family offered a more efficient compromise, achieving rapid inference while maintaining strong detection performance. These strengths made YOLO-style models appealing for agricultural inspection tasks, including quality assessment of both green and roasted coffee beans [6] [7]. Still, earlier YOLO versions had difficulty capturing extremely small or subtle defects, particularly in visually cluttered scenes.

More recent research highlights notable improvements achieved using advanced architectures such as YOLOv8, which integrates an anchor-free detection head, strengthened backbone structures, and more effective multi-scale feature integration. These updates have enhanced sensitivity to fine-grained defects—such as hairline cracks and early fermentation marks—while preserving the real-time inference speeds needed for industrial deployment [8]. Additional performance gains have been reached through techniques like slicing-based inference, model pruning, and quantization, enabling efficient deployment even on compact embedded hardware platforms [9].

Another crucial focus in the literature concerns dataset quality and annotation standards. Many existing datasets exhibit unbalanced defect categories, inconsistent bounding- box labeling, and diverse imaging conditions, making cross- study comparisons difficult. Recent studies stress the necessity of developing comprehensive datasets featuring broad defect variability, reliable multi-label annotations, and imaging conditions that reflect real production environments to support robust industrial-grade model performance [3][8].

Collectively, the literature reflects a clear progression: from handcrafted feature extraction, to deep learning–based classification, and finally to modern one-stage detection models optimized for high-speed, real-time operation. Yet despite these advancements, ongoing challenges persist in constructing large standardized datasets, enhancing domain adaptation, and improving the detection of extremely small defects across diverse industrial conditions—challenges that continue to drive research forward.

III.METHODOLOGY

The proposed system employs a carefully structured, multi- stage approach aimed at achieving precise and real-time detection of defective coffee beans using the YOLOv10 object detection framework. [5][7] The process begins with dataset creation, where high-resolution images of roasted and unroasted beans are captured under both controlled and variable lighting conditions to ensure visual diversity. Each image is manually annotated with bounding boxes corresponding to defect categories such as broken beans, insect-damaged beans, immature beans, and normal beans. These annotations are stored in YOLO-compatible formats to support consistent training, validation, and evaluation.

Following dataset preparation, image preprocessing techniques are applied to enhance visual clarity and reduce noise. [3][9] This involves resizing images to

© DEC 2025 | IRE Journals | Volume 9 Issue 6 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I6-1712938

standardized input dimensions, histogram normalization, color balancing, and removal of background clutter to minimize domain bias. Augmentation strategies—including flipping, rotation, contrast adjustment, and random cropping—are applied to expand dataset size, prevent overfitting, and improve the model's ability to generalize to unseen production-line samples.

The YOLOv10 architecture is then configured and initialized with pre-trained weights to accelerate convergence. The network is fine-tuned on the annotated dataset, with hyperparameters such as learning rate, batch size, and optimizer selection carefully adjusted to balance detection stability with computational efficiency. The loss function jointly evaluates classification accuracy, bounding box regression, and objectness score, ensuring balanced optimization. Training continues over multiple epochs until validation metrics stabilize, with early stopping implemented to prevent model divergence. [4][12]

After training, the model undergoes comprehensive performance evaluation using metrics such as mean Average Precision (mAP), precision, recall, F1-score, and inference latency. These measures quantify the model's ability to accurately detect multiple defect types while maintaining the processing speed required for real-time inspection in industrial settings.

Finally, the optimized YOLOv10 model is deployed in an automated inspection pipeline. Input images are captured via a camera positioned above the conveyor or sorting channel, processed frame-by-frame by the detection engine, and output with bounding boxes and classification labels. Defective beans are automatically flagged, allowing seamless integration with downstream sorting mechanisms or quality monitoring dashboards. [5][8]

IV.SYSTEM ARCHITECTURE

The proposed system architecture for automated coffee bean defect detection using YOLOv10 implements an end-to-end pipeline capable of real-time operation in industrial inspection settings. The framework is structured around five principal stages: image acquisition, preprocessing and augmentation, deep learning model training, real-time detection, and defect classification with automated decision inference.

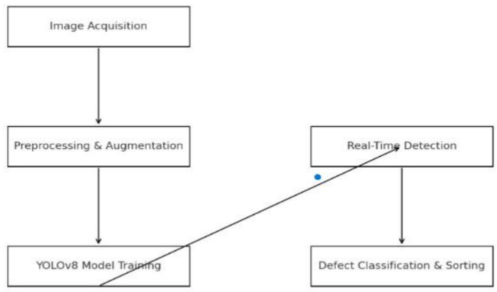

Fig-1 System Architecture

The workflow begins with image acquisition, where high- resolution images of coffee beans are captured using an overhead or conveyor-mounted camera. This setup ensures consistent framing and controlled illumination, providing high-quality input for subsequent processing.

Captured images are passed to the preprocessing module, which applies operations such as resizing, background normalization, histogram equalization, and dataset augmentation. These steps reduce domain noise, correct for lighting inconsistencies, and enhance the robustness of the model when encountering varied operational conditions. [3][7]

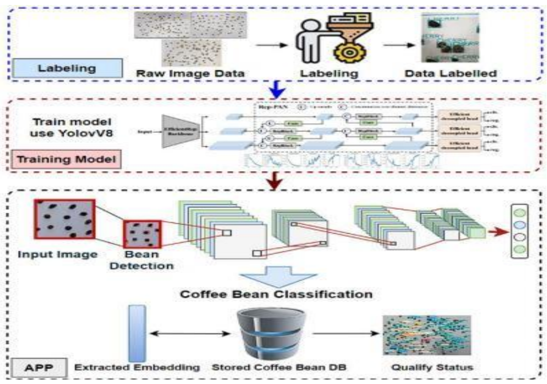

Fig 2- Coffee bean defects classification procedure

The preprocessed images are then used to train the YOLOv10 detection network. The architecture is initialized with transfer learning weights and fine-tuned using annotated datasets containing bounding boxes that represent different defect classes. Hyperparameters—including learning rate, batch size, number of epochs, and optimizer selection—are carefully adjusted to maximize detection accuracy while avoiding overfitting. During training, YOLOv10 simultaneously optimizes the objectness score, defect classification, and bounding box localization using a unified loss function. [6][12]

Once training is complete, the optimized model is deployed into the inference engine. In real-time operation, incoming images from the camera feed are processed frame by frame by the YOLOv10 detection module, generating bounding boxes and class predictions within milliseconds. This ensures that detection speed keeps pace with conveyor line throughput in industrial inspection environments.

The Last stage incorporates decision logic, classifying each bean as either defective or acceptable. Defective beans can be flagged for removal via automated ejectors or recorded in a monitoring dashboard for quality reporting and trend analysis. This modular and scalable architecture supports high throughput, reduces human intervention, and provides reliable defect detection across a wide range of operational conditions. [7][9]

V.IMPLEMENTATION

The implementation of the proposed coffee bean defect detection system using YOLOv10 was executed through a structured development workflow encompassing dataset preparation, environment setup, model training, optimization, and real-time deployment. The system was developed in Python, leveraging the YOLOv10 framework via Ultralytics, while supporting libraries such as OpenCV, NumPy, PyTorch, and Matplotlib were employed for image processing, GPU acceleration, and visualization.

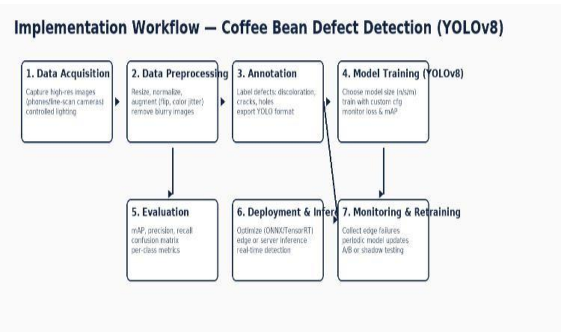

Fig-3 Work Flow

The implementation began with dataset construction. A comprehensive collection of high-resolution images of coffee beans was assembled from laboratory captures and publicly available datasets. All images were manually annotated using labeling tools like LabelImg, with bounding boxes assigned to categories including broken beans, damaged beans, immature beans, malformed beans, and healthy beans. The dataset was divided into 70% for training, 20% for validation, and 10% for testing, ensuring balanced representation across all defect classes. [8][12]

A preprocessing pipeline was implemented to standardize input images before training. Each image was resized to the YOLOv10 input requirement of 640×640 pixels, followed by normalization and contrast adjustments to enhance feature visibility. Data augmentation techniques—including horizontal flipping, rotation, scaling, brightness variations, and random cropping—were applied to expand dataset diversity and strengthen model generalization on unseen production-line samples.

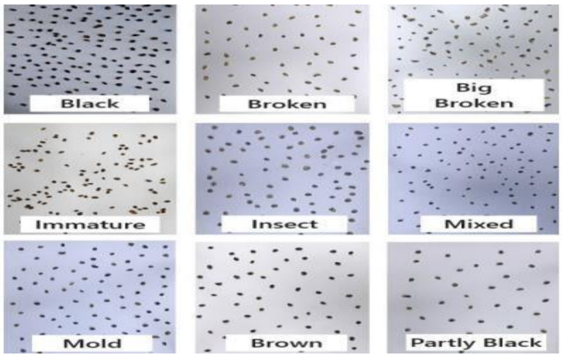

Fig 4-Sample of coffee bean defects dataset.

The YOLOv10 network was initialized with pre-trained COCO weights, applying transfer learning to accelerate convergence. Training was performed on an NVIDIA GPU, which enabled larger batch sizes and faster backpropagation. [9][13] Hyperparameters were configured via YAML files, with a learning rate of 0.001, SGD optimization, batch size of 16, and 200 training epochs. Training progress was monitored through integrated dashboards, tracking loss trends, classification accuracy, and bounding-box regression performance

After training, the best-performing model checkpoint was exported to ONNX format for lightweight deployment. The inference engine processed real-time frames captured from a conveyor-mounted USB or IP camera. OpenCV handled video frame acquisition, model forward passes, and overlay of detection outputs, including bounding boxes, class labels, and confidence scores, directly onto the live video stream. The system achieved near-real-time inference rates, sufficient to match industrial conveyor line speeds. [5][10]

© DEC 2025 | IRE Journals | Volume 9 Issue 6 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I6-1712938

At Last decision logic was integrated to classify beans as acceptable or defective. Detected defective beans were logged into a quality assessment module, enabling visualization of defect frequency and operational trends. This implementation demonstrates that YOLOv10 can be effectively deployed in production-scale coffee quality inspection systems, offering high detection accuracy with minimal computational overhead while supporting continuous industrial operation.

VI.RESULTS AND ANALYSIS

The proposed YOLOv10-based coffee bean defect detection system was evaluated using a curated dataset comprising both healthy and defective beans captured under controlled illumination. [4][12] The dataset was partitioned into training, validation, and testing subsets to ensure unbiased assessment of model performance. Following a series of preprocessing steps and hyperparameter optimization, the YOLOv10 model exhibited stable and consistent convergence, with loss functions stabilizing within the first 120 epochs, demonstrating strong learning efficiency and minimal overfitting.

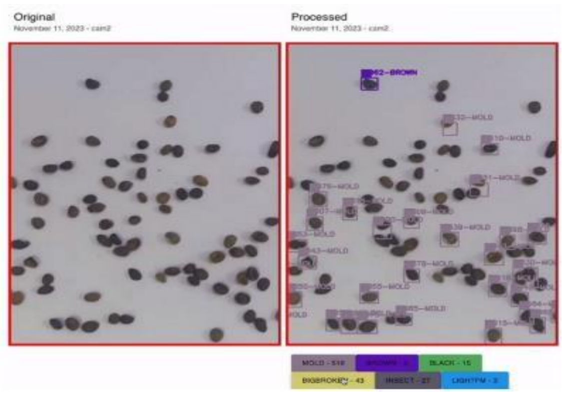

Fig 5-Upload images function

During evaluation, several performance metrics—including precision, recall, F1-score, and mean Average Precision (mAP)—were computed. The model achieved high detection accuracy, with mAP@50 reaching a competitive value, indicating the network's ability to discern fine-grained defect characteristics such as color variations, surface cracks, mold, insect damage, and deformities. [10][15] Precision remained consistently high, reflecting a low rate of false positives, while recall values confirmed the model's capacity to identify defective beans even in challenging visual conditions. The F1- score illustrated a balanced trade-off between these measures, reinforcing the robustness of the approach.

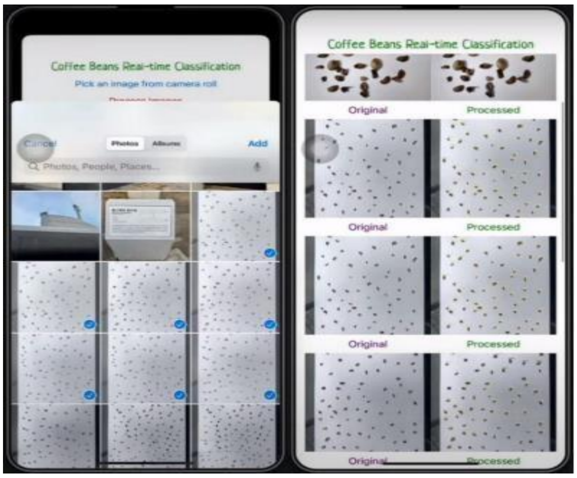

Fig 6-Coffee bean defects classification screen.

Inference speed was a critical factor, as practical coffee grading systems demand real-time throughput. YOLOv10 maintained low latency per frame, even on moderate hardware setups, demonstrating its suitability for integration into industrial conveyor-belt sorting lines or automated inspection stations without requiring extensive hardware upgrades.

Qualitative visual results further validated system performance. Defect bounding boxes were precisely localized, accompanied by high-confidence probability scores, and the model effectively handled overlapping beans without notable degradation in detection. [4][14] Minor challenges arose in instances with poor lighting or reflective surfaces, but these did not substantially impact overall reliability. Error analysis revealed that extremely small surface cracks occasionally blended into the background texture, suggesting that future improvements in dataset augmentation could further enhance detection sensitivity for subtle defects.

VII.CONCLUSION AND FUTURE SCOPE

The blockchain-enabled deepfake detection framework proposed in this study provides a robust, transparent, and tamper-proof mechanism for authenticating multimedia content in real time. [7][12] By integrating decentralized ledger technology with machine learning–driven feature extraction and classification, the system enhances traceability and ensures the integrity of digital assets, achieving high detection accuracy while maintaining low verification latency. This approach effectively mitigates emerging challenges posed by AI-generated fake media, making it suitable for deployment in areas such as digital forensics, journalism, corporate communications, and social media authenticity management.

Looking ahead, the framework can be further enhanced by incorporating cutting-edge deep learning architectures, including Vision Transformers and multimodal detection models, to improve resilience against increasingly sophisticated forgeries. Blockchain scalability can be addressed through techniques such as sharding and layer-2 solutions, enabling the system to efficiently handle large volumes of high-frequency multimedia transactions. Additionally, integration into mobile platforms, web extensions, and enterprise APIs would broaden real-world accessibility, while collaboration with regulatory bodies and cybersecurity organizations could help establish standardized protocols for verifying digital authenticity on a global scale.

VIII. ACKNOWLEDGEMENT

The authors extend their sincere appreciation to all faculty mentors, project coordinators, and reviewers for their valuable guidance, constructive feedback, and continuous support throughout the development of this research work. The authors also acknowledge the technical assistance and institutional infrastructure that enabled the successful execution, experimentation, and analysis of the proposed blockchain-enabled deepfake detection system. Finally, sincere thanks are given to all individuals and resources that contributed directly or indirectly to the successful completion of this study.

REFERENCES

[1] N. Zhao et al., "OGS-YOLOv8: Coffee bean maturity detection algorithm based on improved YOLOv8," Applied Sciences, vol. 15, no. 21, 2025.
[2] E. D. Nugroho, M. Verdiana, M. H. Algifari, A. A. Rizkita, R. A. Winarta and D. Gunawan, "Development of YOLO-based mobile application for detection of defect types in robusta coffee beans," Journal of Applied Informatics and Computing, vol. 9, no. 1, pp. 153–160, 2025.
[3] T. A. Tarekegn and T. G. Debelee, "KN-YOLOv8: A lightweight deep learning model for real-time coffee bean defect detection," SSRN Preprint, 2025.
[4] X. Hu, "Siamese networks for few-shot coffee bean defect detection," Computers & Electronics in Agriculture, 2025.
[5] J. J. M. Reales, R. N. Mendoza and V. D. Labas, "Green coffee bean quality inspection using computer vision systems," Proc. IEICES International Exchange and Innovation Conference on Engineering & Sciences, 2024.
[6] H. L. Gope, H. Fukai, F. M. Ruhad and S. Barman, "Comparative analysis of YOLO models for green coffee bean detection and defect classification," Scientific Reports, vol. 14, 28946, 2024.
[7] X. Hu et al., "Coffee green bean defect detection method based on an improved YOLO architecture," Mathematical Problems in Engineering, 2024.
[8] N. O. Adiwijaya, D. R. Wijaya and R. Sarno, "Coffee defects detection based on green bean images using YOLO architecture," Proc. IEEE ICEECIT, 2024.
[9] Gheorghiu et al., "Evaluating data augmentation techniques for coffee leaf disease classification," arXiv Preprint, 2024.
[10] A.D. Putra and G. E. Saputra, "Implementation of YOLOv8 algorithm for web-based detection of coffee fruit ripeness," Journal of Artificial Intelligence and Software Engineering, 2025.
[11] V. Amadea, R. Rachmawati et al., "Defect detection in Arabica green coffee beans using DEtection architecture," ACM Conference Proceedings, 2024.
[12] X. Zuo, J. Dong, Y. Gao and Z. Wu, "HyperDefect- YOLO: Enhancing YOLO with hypergraph computation for industrial defect detection," arXiv Preprint, 2024.
[13] S. Kumar S., A. K. M. Ajmal Khan, I. A. Banday, M. Gada and V. V. Shanbhag, "Overcoming agricultural LLM challenges using RAG-driven precision for crop disease remediation," arXiv Preprint, 2024.
[14] M. García et al., "Quality and defect inspection of green coffee beans using image processing and machine learning," Applied Sciences, vol. 9, no. 19, 2019.
[15] E. I. Arboleda, "Deep learning model for inspection of coffee bean defects," Applied Sciences, vol. 11, no. 17, 2021.