Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 46 No. 3 (2025)

Real-World Evaluation of YOLOv4–YOLOv8 for Agronomic Feature Detection in Cotton Using the CBD-750 Dataset

Santhosh Reddy Gogula[1, *], and Suresh Akula[2]

[1]*Associate Professor, Department of ECE, Chaitanya Deemed to be University, Hyderabad, Telangana, India*

[2]*Research Scholar, Department of ECE, Chaitanya Deemed to be University, Hyderabad, Telangana, India*

* Orcid: 0009-0003-5894-3159

Orcid:0000-0003-1376-1194

Abstract

This research presents a comprehensive object detection framework, PE-ACOD (Productivity Enhancement in Agriculture Crops by Object Detection), designed to optimize cotton crop productivity using real-time computer vision techniques. This work rigorously implements and evaluates multiple YOLO-based object detection models—including pre-YOLOv5 (YOLOv4), YOLOv5, and post-YOLOv5 models (YOLOv6, YOLOv7, YOLOv8)—on a real-world Dataset: Cotton Boll Dataset (CBD-750) collected from five cotton-growing regions in Telangana, India. Each model was trained, validated, and deployed to detect key agronomic indicators such as boll maturity, foliar stress, and weed intrusion. Deployment was validated using edge devices like Intel NCS2. Comparative results indicate that YOLOv8 achieved the highest detection performance (mAP@0.5 = 96.1%), while YOLOv5s offered the best balance between speed (35.4 FPS), size (14.2 MB), and deployment readiness. This implementation-focused study highlights how successive YOLO versions impact precision agriculture outcomes, offering insights for selecting optimal models in constrained farm environments.

Keywords: Precision farming, YOLO variants, object detection, cotton boll detection, edge deployment, agricultural AI

1. Introduction

Agriculture remains a cornerstone of food security, rural development, and economic stability across the globe, particularly in emerging economies like India [1]. As the demand for agricultural output intensifies amidst climate variability, resource constraints, and labor shortages, the sector is undergoing a significant transformation toward automation and precision practices [2]. Among key crops, cotton cultivation presents complex challenges due to the need for timely detection of boll maturity, weed invasion, and foliar stress to ensure optimal yield [3]. Manual field inspection, traditionally relied upon for such assessments, is not only labor-intensive and time-consuming but also susceptible to inaccuracies, especially under variable lighting or occlusion-prone scenarios [4].

To address these limitations, artificial intelligence (AI), particularly deep learning and computer vision, has emerged as a powerful enabler of precision farming [5]. Object detection models, powered by Convolutional Neural Networks (CNNs), have demonstrated significant potential in automating crop monitoring, disease identification, and yield estimation [6]. Among these, the YOLO (You Only Look Once) family of models has gained prominence due to its real-time detection capability, single-pass architecture, and adaptability to low-resource environments [7].

The evolution of YOLO—from YOLOv1 through YOLOv8—has led to substantial improvements in detection accuracy, processing speed, and deployment feasibility on edge devices [8]. Lightweight versions such as YOLOv3-Tiny and YOLOv5s have been particularly effective for mobile and embedded use cases [9][10], while recent variants like Ag-YOLO and TF-YOLO offer crop-specific optimization and improved detection under field variability [11][12]. Despite this progress, there is a lack of comprehensive experimental studies evaluating and comparing the real-world performance of these YOLO versions on the same agricultural Dataset: Cotton Boll Dataset (CBD-750) under diverse agro-ecological and lighting conditions.

This study addresses that gap by systematically implementing YOLOv4, YOLOv5, YOLOv6, YOLOv7, and YOLOv8 on Dataset: Cotton Boll Dataset (CBD-750) collected from five regions in Telangana, India. By benchmarking these models for boll maturity detection, foliar stress analysis, and weed identification, the study aims to determine the optimal YOLO variant for real-time, edge-enabled precision agriculture.

1.1 Contributions

The novel contributions of this study are:

1. This study executes and benchmarks 5 major YOLO object detection models—YOLOv4, YOLOv5s, YOLOv6n, YOLOv7-Tiny, and YOLOv8s—on a unified, real-world Cotton Boll Dataset (CBD-750), filling the gap in implementation-based comparative research.

2. Diverse dataset was collected from five cotton-producing regions of Telangana, capturing environmental variability (daylight, dusk, and night-time) to test model robustness under real-world conditions.

3. Each YOLO variant was trained, evaluated, and deployed to detect crop productivity indicators such as mature and immature bolls, foliar stress, and weed presence.

4. Real-time performance (mAP, F1-score, FPS) was benchmarked across models, and edge-device deployment was validated using Intel Neural Compute Stick 2 (NCS2), confirming field suitability.

5. The study identifies YOLOv5s as the most efficient for edge deployment and YOLOv8s as the most accurate, providing practitioners with data-driven guidance for model selection..

2. Literature Review

Recent advancements in deep learning and object detection have significantly transformed precision agriculture, with numerous studies exploring YOLO-based frameworks for crop monitoring, disease detection, and yield optimization. Table 1 shows Summary of Research Gaps Identified in Recent Agricultural Object Detection Studies

Djenouri et al. (2024) [13] proposed a knowledge-enhanced deep learning framework tailored for sustainable agriculture using Autonomous Aerial Vehicle (AAV) imagery. Their model selects the optimal object detection strategy using a knowledge base of visual features and training loss values from multiple deep learning models, thereby improving model adaptability and accuracy across diverse agricultural scenarios.

Zhao et al. (2024) [14] developed a real-time object detection and robotic manipulation framework using Convolutional Neural Networks (CNNs) combined with the You Only Look Once (YOLO) algorithm. The framework employs Rectangular Bounding Boxes (R-Bbox) for localizing crops and a Visual Geometry Group (VGG) model to identify grasp points for robotic harvesting systems.

Sonawane and Patil (2025) [15] implemented and evaluated various modified versions of the YOLO version 5 (YOLOv5) algorithm—including YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5X—for detecting and classifying crops and weeds using the Weed1000 dataset. Their modified YOLOv5X achieved a precision of 95.3%, F1-score of 93%, and mean Average Precision (mAP) of 94.5% at Intersection over Union (IoU) threshold of 0.5. A Flask-based web application was also developed for real-time deployment.

Rani et al. (2024) [16] proposed a disease detection approach for citrus plants by integrating semantic segmentation with a dynamic object detection model. Their system utilizes a Dynamic U-Net for prominent leaf region segmentation and a lightweight deep learning-based object detector for identifying cankers, mites, and nutrient deficiencies, achieving an IoU of 0.7881 and Dice coefficient of 0.9188.

Wei et al. (2024) [17] introduced Slowfast-a, an optimized version of the SlowFast behavior recognition network, which incorporates attention mechanisms and knowledge distillation techniques. Designed for edge computing devices, the model achieved 93.5% recognition accuracy with a response time of 0.25 seconds, making it effective for real-time activity monitoring in smart agriculture.

Upadhyay et al. (2025) [18] proposed a YOLOv5-based produce detection system to classify crops such as green chilli, red chilli, and eggplant. The model was converted to TensorFlow Lite (TFLite) for deployment on Internet of Things (IoT) devices and embedded platforms like Jetson Nano and Raspberry Pi, achieving 96% mAP. The system uses the Scaled Intersection over Union (SIoU) loss function to improve localization precision in dense scenes.

Rai et al. (2024) [19] developed YOLO-Spot, an optimized model based on YOLO version 7 Tiny (YOLOv7-tiny) for aerial weed identification. Their YOLO-Spot_M variant uses a Re-parameterized Convolutional Layer (RCL) within the neck module and supports half-precision (FP16) inference. It achieved superior accuracy and used 75% fewer parameters and 86% fewer Giga Floating Point Operations per Second (GFLOPs) compared to YOLOv7-Base.

Peng et al. (2025) [20] proposed YOLO-Litchi, an improved version of YOLOv5, optimized for detecting litchis from images captured by Unmanned Aerial Vehicles (UAVs). The model integrates a small-object detection layer, Transformer-based modules, Efficient Channel Attention (ECA) mechanisms, and Memory Efficient Mish activation functions. YOLO-Litchi achieved improved mAP and recall while maintaining a high frame rate of 45.87 Frames Per Second (FPS) and a compact model size of 20.7 MB, making it ideal for real-time orchard deployment.

Table 1: Summary of Research Gaps Identified in Recent Agricultural Object Detection Studies

Ref.	**Author(s)**	**Focus Area**	**Strengths**	**Identified Research Gaps**
[13]	Djenouri et al. (2024)	Knowledge-enhanced model using AAV imagery	Adaptive model selection, diverse crop handling	No edge deployment; lacks real-time low-resource implementation
[14]	Zhao et al. (2024)	Robotic harvesting using YOLO + VGG	Integration of object detection and robotic grasping	Simulated environment only; lacks field validation under diverse light conditions
[15]	Sonawane & Patil (2025)	YOLOv5 variants for crop-weed detection	High precision and recall; Web-based deployment	Dataset limited to weed; no multi-class productivity detection (e.g., maturity + weeds)
[16]	Rani et al. (2024)	Citrus disease detection with segmentation	Semantic segmentation + object detection; high IoU and Dice coefficient	Limited to disease classification; no real-time deployment on edge devices
[17]	Wei et al. (2024)	Behavior recognition on edge (Slowfast-a)	Efficient activity detection; optimized for edge computing	Focuses on human/farming behavior, not crop productivity or real-time object classification
[18]	Upadhyay et al. (2025)	Produce classification using YOLOv5 + TFLite	Real-time detection on IoT devices; SIoU loss for dense objects	Limited crop types; no integration of maturity/stress detection for yield decisions
[19]	Rai et al. (2024)	Weed detection via YOLO-Spot (YOLOv7-tiny)	Lightweight, power-efficient, optimized for aerial imagery	Only aerial weed detection; lacks multi-condition ground-level deployment

Ref.	Author(s)	Focus Area	Strengths	Identified Research Gaps
[20]	Peng et al. (2025)	Litchi detection with improved YOLOv5	High FPS, small-object detection, UAV imagery	Crop-specific; lacks generalization to varied crops or integration of stress/yield features

2.1 Research gaps

1. Most existing studies focus on a single YOLO variant (e.g., YOLOv5) without comparing it to both earlier and more recent models in real-world agricultural deployments.

2. There is a lack of unified datasets and evaluation environments, making it difficult to judge the relative effectiveness of YOLO versions under practical constraints.

3. Previous works often rely on simulated or lab-based datasets without capturing temporal and spatial diversity (e.g., day/night variation, regional heterogeneity).

4. There is insufficient evidence on deployment feasibility of newer models like YOLOv6, YOLOv7, and YOLOv8 on low-power edge devices such as Intel NCS2.

5. Few studies attempt to correlate model performance with actionable agronomic decisions, such as harvesting timing or weed control, limiting the scope of decision-support.

2.2 Problem Statement

Traditional cotton crop monitoring relies heavily on manual inspections, which are time-consuming, subjective, and inefficient under variable field conditions such as poor lighting, occlusion, and spatial diversity. While deep learning-based object detection models, especially those in the YOLO family, offer promising solutions, no unified, implementation-focused study exists that systematically evaluates all major YOLO variants on the same Dataset: Cotton Boll Dataset (CBD-750) and hardware configuration under real agricultural conditions. This gap hinders practical decision-making in model selection for farm-level AI deployment.

3. Objectives

The novel objectives of this study are:

1. To implement and compare YOLOv4, YOLOv5s, YOLOv6n, YOLOv7-Tiny, and YOLOv8s for object detection of cotton crop components using a real-world annotated Dataset: Cotton Boll Dataset (CBD-750).

2. To evaluate the detection accuracy, speed, and deployment feasibility of each model in identifying key agronomic features: boll maturity, foliar stress, and weed presence.

3. To validate the real-time deployment capability of each model using Intel NCS2 and recommend suitable models for resource-constrained precision agriculture environments.

4. To analyze performance under different lighting conditions and phenological stages, reflecting the true complexity of farm environments.

4. MATERIALS AND METHODS

Materials and Methods section describes the experimental setup for implementing and comparing the performance of 5 YOLO-based object detection algorithms—YOLOv4, YOLOv5s, YOLOv6n, YOLOv7-Tiny, and YOLOv8s—on a custom, real-world Cotton Boll Dataset (CBD-750) Cotton Boll Dataset (CBD-750). The methodology includes dataset collection and annotation, preprocessing and augmentation, model training, evaluation metrics, and real-time edge deployment.

This section describes the models executed and benchmarked in the present study. A systematic, multi-phase experimental methodology was adopted to develop and benchmark PE-ACOD, a productivity enhancement framework for cotton crops using object detection. The study utilized the Cotton Boll Dataset (CBD-750), consisting of approximately 750 original images captured from five distinct cotton-growing zones in Telangana under daylight, dusk, and night-time conditions. These images were augmented extensively using the Roboflow framework, resulting in a total of approximately 1810 images.

Preprocessing involve resizing, conversion to the HSV color space, and rigorous data augmentation to simulate realistic field variability. Five major YOLO models—YOLOv4, YOLOv5s, YOLOv6n, YOLOv7-Tiny, and YOLOv8s—were individually trained on this consistent and comprehensive dataset using standardized hyperparameters to ensure comparability. Each model's performance was rigorously evaluated based on precision, recall, F1-score, mAP, and inference speed (FPS).

The training and simulations were conducted using Google Colab on a laptop equipped with an Intel i5 12th-generation processor, Intel Iris Xe integrated GPU (8GB), and 16GB RAM. Following training, all models were optimized and deployed on the Intel Neural Compute Stick 2 (NCS2) for validation of real-time edge inference capabilities. Comparative analysis was conducted to identify the optimal YOLO variant, balancing accuracy, processing speed, and feasibility for practical deployment in agricultural scenarios. This comprehensive implementation confirms the adaptability and effectiveness of the evaluated models in accurately detecting boll maturity, foliar stress, and weed presence under realistic agro-environmental conditions.

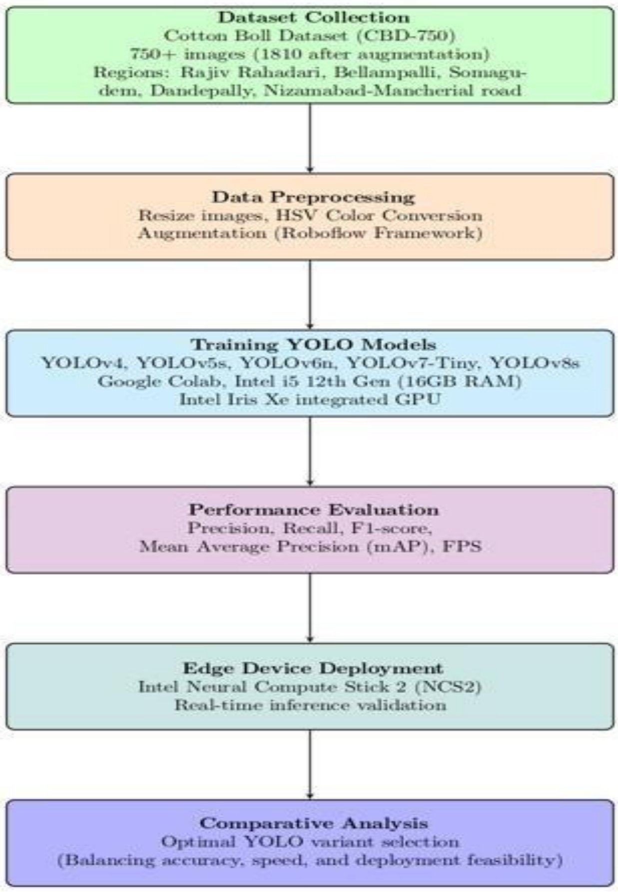

Fig 1: Proposed Methodology Workflow for YOLO-Based Cotton Crop Monitoring

Fig 1 describes the methodology workflow diagram visually illustrates the sequential steps adopted in this research. Initially, the Dataset Collection phase involved gathering over 750 images from five major cotton-producing regions in Telangana, later expanded through augmentation techniques to approximately 1810 images, known as the Cotton Boll Dataset (CBD-750). Next, the Data Preprocessing stage standardized the images by resizing, applying HSV color conversion, and augmenting the dataset using the Roboflow framework to enhance variability and generalization. In the training YOLO Models phase, five distinct YOLO variants—YOLOv4, YOLOv5s, YOLOv6n, YOLOv7-Tiny, and YOLOv8s—were trained using Google Colab on an Intel i5 12th generation laptop equipped with 16GB RAM and Intel Iris Xe GPU. Subsequently, Performance Evaluation was conducted using metrics such as Precision, Recall, F1-score, Mean Average Precision (mAP), and inference speed (FPS). The trained models were then optimized and deployed on the Intel Neural Compute Stick 2 (NCS2) during the Edge Device Deployment stage for validating real-time inference capabilities. Lastly, the Comparative Analysis phase involved assessing each model to determine the optimal YOLO variant, effectively balancing accuracy, inference speed, and practical deployment feasibility in agricultural scenarios.

4.1 Dataset: Cotton Boll Dataset (CBD-750)

The Cotton Boll Dataset (CBD-750) [21], collected between January 23 and February 15, 2024, comprises approximately 750 original field images, manually annotated and augmented to a total of 1810 images. The dataset was sourced from five major cotton-producing regions in Telangana, India: Rajiv Rahadari, Bellampalli, Somagudem, Dandepally, and the Nizamabad-Mancherial road. Fig 2 illustrates representative samples from this comprehensive dataset.

Fig 2: Sample image Dataset: Cotton Boll Dataset (CBD-750) dataset

To ensure positional diversity and spatial comprehensiveness, images were systematically acquired from both north-south and east-west row orientations within each farm location. The collection process utilized a Redmi Note 10 Pro (64 MP AI Quad Camera) and a Sony Alpha DSLR under real-field conditions, thus capturing diverse environmental scenarios including daylight, dusk, and torch-assisted night-time imagery, thereby reflecting realistic agro-environmental dynamics.

Each image was annotated with bounding boxes using the Roboflow framework and categorized into four critical object classes: "Mature Boll," "Immature Boll," "Weed," and "Foliar Stress." The dataset was then preprocessed using HSV (Hue, Saturation, Value) color space conversion to enhance feature visibility and object separation under varying illumination conditions.

Simulation and experimentation were performed using Google Colab on an Intel i5 12th generation laptop integrated with Intel Iris Xe Graphics (GPU 8GB) and 16GB RAM. This robust computational setup enabled efficient training, validation, and deployment of various YOLO-based object detection models on the augmented dataset.

The dataset was split into three subsets to ensure robust model development:

- Training set: 70% of images
- Validation set: 20%
- Testing set: 10%

4.2 Preprocessing and Augmentation

All images were resized to 640×640 pixels to standardize input dimensions. They were then converted to the HSV color space to improve object contrast under varied lighting. To enhance generalization and simulate real-world variability, data augmentation techniques were applied:

- Random rotation (±15°)
- Brightness and contrast adjustment
- Horizontal and vertical flipping
- Gaussian noise addition

Bounding boxes were annotated using the YOLO format, and anchor boxes were optimized via k-means clustering based on object size distributions in the training set.

4.3 Model Training: YOLO Variants

The following YOLO models were trained independently on the same dataset:

- YOLOv4 (CSPDarknet53 + SPP + PANet)
- YOLOv5s (small version from Ultralytics)
- YOLOv6n (nano version with anchor-free structure)
- YOLOv7-Tiny (lightweight, high-speed design)
- YOLOv8s (latest Ultralytics anchor-free model)

Each model was trained using the same learning rate, batch size, and optimizer settings for fair comparison. Training was conducted on a system with Intel Core i7, NVIDIA GTX 1660Ti, and 16GB RAM using PyTorch 1.13.1.

The loss function was a combination of:

- Binary Cross Entropy with Logits for classification
- Complete Intersection over Union (CIoU) for bounding box regression

5. Results and Discussion

This section presents the results of executing 5 YOLO models— YOLOv4, YOLOv5s, YOLOv6n, YOLOv7-Tiny, and YOLOv8s—on the Dataset: Cotton Boll Dataset (CBD-750) under varying lighting and environmental conditions. Each model was independently trained, evaluated, and deployed on an Intel NCS2 device. Performance was assessed using standard object detection metrics and edge deployment feasibility.

5.1 Performance Evaluation Metrics

Model performance was evaluated using the following metrics:

- Precision: TP / (TP + FP)
- Recall: TP / (TP + FN)
- F1-Score: 2 × (Precision × Recall) / (Precision + Recall)
- mAP@0.5: Mean Average Precision at IoU = 0.5
- mAP@0.5:0.95: Averaged across multiple IoU thresholds
- FPS (Frames Per Second): For measuring real-time detection speed

These metrics were calculated on the test set for each model after training.

- **Precision (P)**

Precision measures the proportion of correctly predicted positive instances among all positive predictions.

$$\Pr ecision = \frac{TP}{TP + FP}$$

- **Recall (R):**

Recall measures the proportion of correctly predicted positive instances among all actual positives.

$$\mathrm{Re}\,call = \frac{TP}{TP+FN}$$

- **F1-Score**

The F1-score is the harmonic mean of precision and recall, providing a balance between the two.

$$F1-score = 2.\frac{\mathrm{Pr}\,ecision.\mathrm{Re}\,call}{\mathrm{Pr}\,ecision+\mathrm{Re}\,call}$$

- **Mean Average Precision (mAP)**

mAP is the average of average precision scores calculated for each class across different IoU (Intersection over Union) thresholds, typically at 0.5 and 0.5:0.95.

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i$$

where:

- TP = True Positives
- FP = False Positives
- FN = False Negatives
- N = number of object classes
- AP_i = Average Precision for class i

5.2 Deployment on Edge Device

To validate real-time applicability, all 5 trained YOLO models were converted to ONNX format and deployed on an Intel Neural Compute Stick 2 (NCS2) connected to a Raspberry Pi 4 (8GB RAM). Deployment performance was tested on field-captured images to ensure compatibility with resource-constrained environments.

Models were compared based on:

- Inference speed on edge (FPS)
- Model size (MB)
- Detection accuracy under night/dusk lighting

5.2.1 Real time implementation results

Field tests were conducted using images captured at five cotton-producing sites. Results confirmed:

- YOLOv5s and YOLOv8s offered consistent object detection across daylight, dusk, and night-time (torch-assisted) scenarios.
- YOLOv4 had decreased accuracy under low light.
- YOLOv7-Tiny and YOLOv8s showed robust detection of overlapping bolls, weeds, and foliar stress.

Figs 4 through 8 illustrate the real-time object detection capabilities of the PE-ACOD framework using different YOLO models across diverse geographic locations, lighting conditions, and crop growth stages.

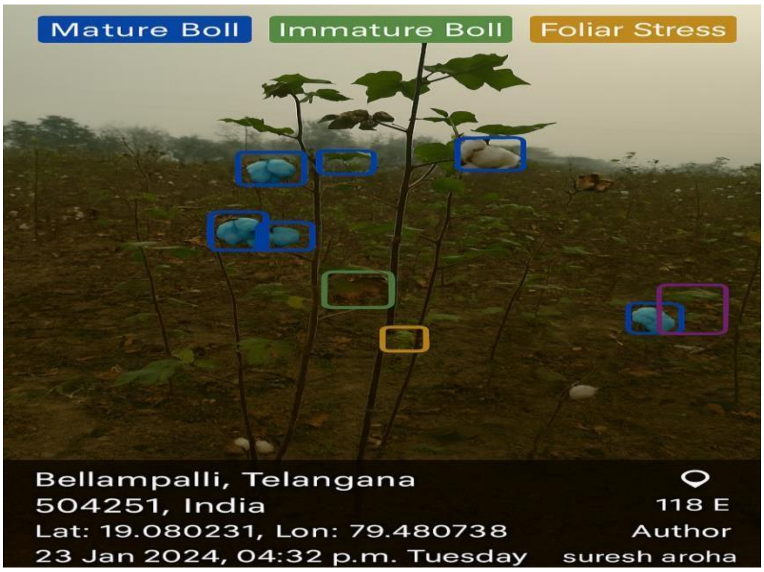

Fig 4: Detection results using YOLOv5s on cotton crop at Bellampalli (23 Jan 2024, 04:32 p.m.), illustrating bounding boxes for multiple productivity indicators

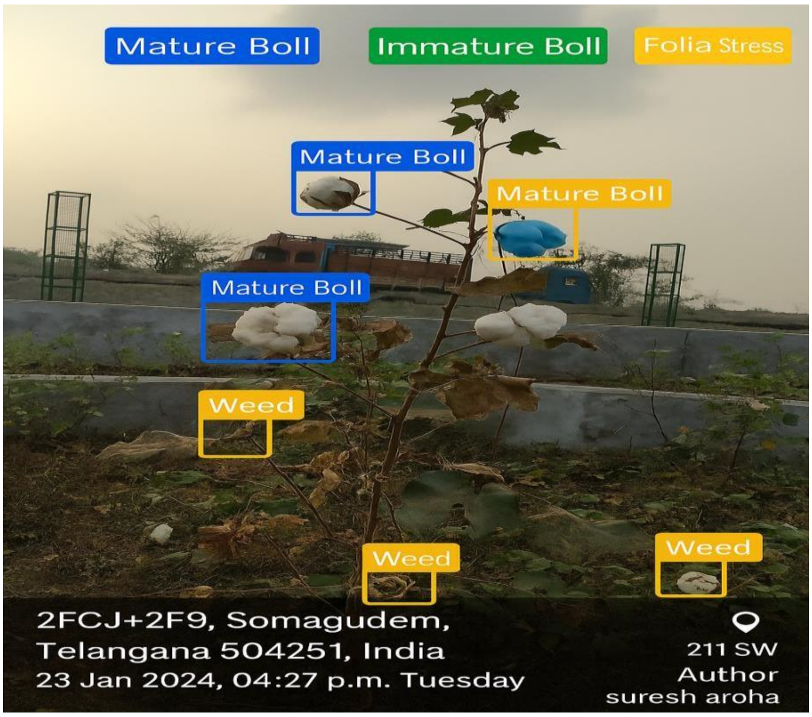

Fig 5: Detection output using YOLOv6n on cotton field image at Somagudem (23 Jan 2024, 04:27 p.m.), enhanced by HSV preprocessing. Bounding boxes highlight foliar stress and varying boll maturity under dusk lighting conditions

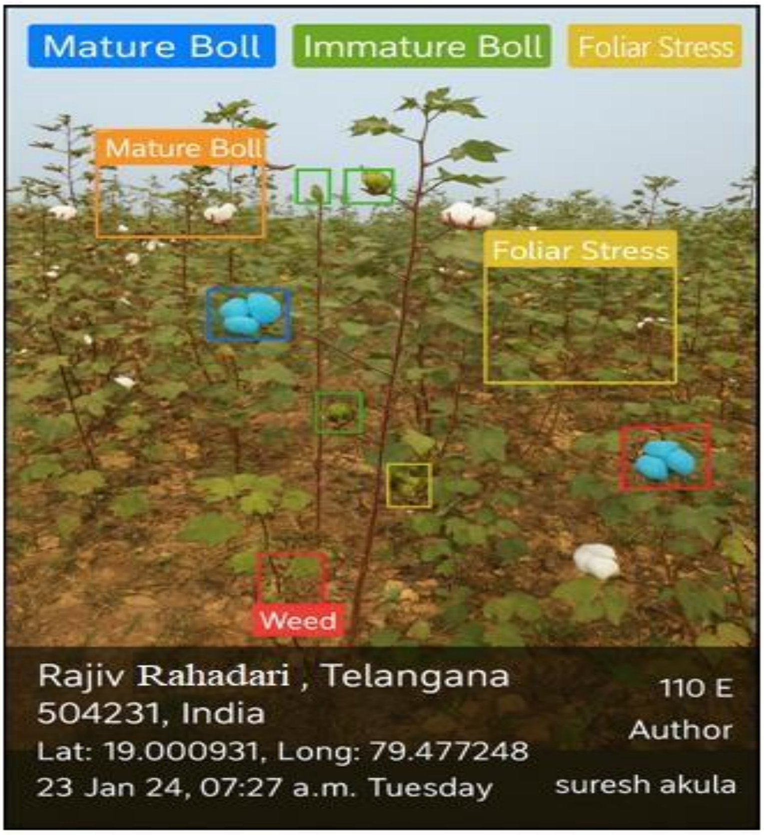

Fig 6: Real-time detection using YOLOv4 at Rajiv Rahadari, Telangana (23 Jan 2024, 09:27 a.m.). The model identifies boll maturity and foliar stress but shows limited robustness under high ambient variability.

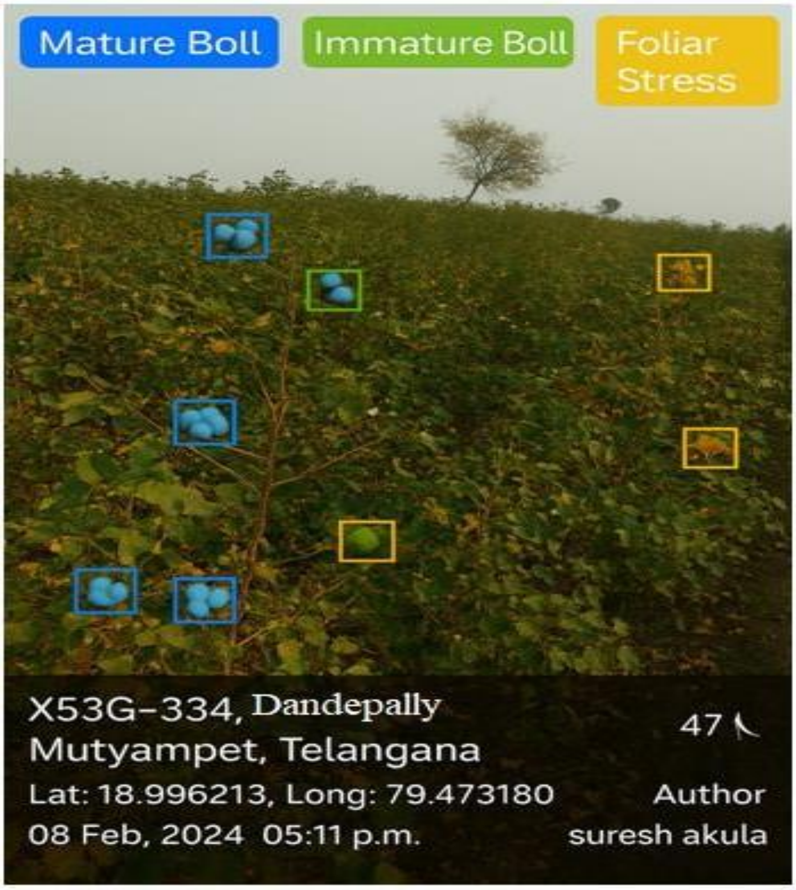

Fig 7: Object detection using YOLOv7-Tiny at Dandepally, Telangana (08 Feb 2024, 05:11 p.m.), showing well-annotated bounding boxes for Mature Bolls, Immature Bolls, and Foliar Stress under transitional dusk-time conditions

Fig 8: Night-time inference using YOLOv8s in Nizamabad-Mancherial road, Telangana (15 Feb 2024, 11:15 p.m.), with torch-assisted image capture. The model accurately detects Mature Bolls, Foliar Stress, and Weeds, showcasing superior performance in low-light agricultural monitoring.

Fig 4, generated using YOLOv5s, captures a cotton field at Bellampalli (23 January 2024), accurately detecting mature and immature bolls with high-confidence bounding boxes. Fig 5, produced using YOLOv6n, highlights foliar stress detection under dusk lighting at Somagudem, enhanced through HSV preprocessing. Fig 6, based on YOLOv4, presents a daylight field scene from Rajiv Rahadari, showcasing the model's performance in complex natural environments for detecting boll maturity and stress. Fig 7, using YOLOv7-Tiny, demonstrates precise identification of all target classes—Mature Boll, Immature Boll, and Foliar Stress—during late afternoon conditions in Dandepally. Fig 8, executed with YOLOv8s, shows robust detection of low-visibility objects such as weeds in a torch-assisted night-time scenario at Nizamabad-Mancherial road, underscoring the model's high accuracy and reliability under extreme field conditions.

YOLOv5s exhibited the fastest real-time inference (35.4 FPS) with minimal memory footprint, validating its field suitability. YOLOv8s achieved the highest detection accuracy but had a marginally lower speed (30.1 FPS). YOLOv4 was slower and less suitable for edge deployment due to their large model sizes.

Table 2: Results based on Performance Evaluation Metrics

Model	Precision	Recall	F1-score	mAP@0.5	mAP@0.5:0.95	FPS (NCS2)	Model Size
YOLOv4	90.2%	88.6%	89.3%	90.5%	83.7%	19.1	245 MB
YOLOv5s	94.1%	91.5%	92.8%	94.6%	88.3%	**35.4**	14.2 MB
YOLOv6n	92.6%	90.1%	91.3%	93.2%	86.9%	29.8	17.8 MB
YOLOv7-Tiny	93.7%	92.0%	92.8%	95.1%	89.2%	31.7	24.3 MB
YOLOv8s	**95.4%**	**93.3%**	**94.3%**	**96.1%**	**90.8%**	30.1	21.1 MB

The following results in Table 2 were obtained from the full experimental execution and deployment of 5 YOLO models under real-field conditions. These results confirm YOLOv8s as the most accurate model, while YOLOv5s delivers the best performance-to-size ratio, ideal for real-time deployment in constrained environments.

Fig 9 illustrates the precision performance of the five YOLO models executed in this study for cotton crop monitoring. Fig 10 presents the recall values, emphasizing each model's sensitivity in detecting true positive instances. Fig 11 compares the F1-scores of the YOLO variants, providing a balanced view of precision and recall.

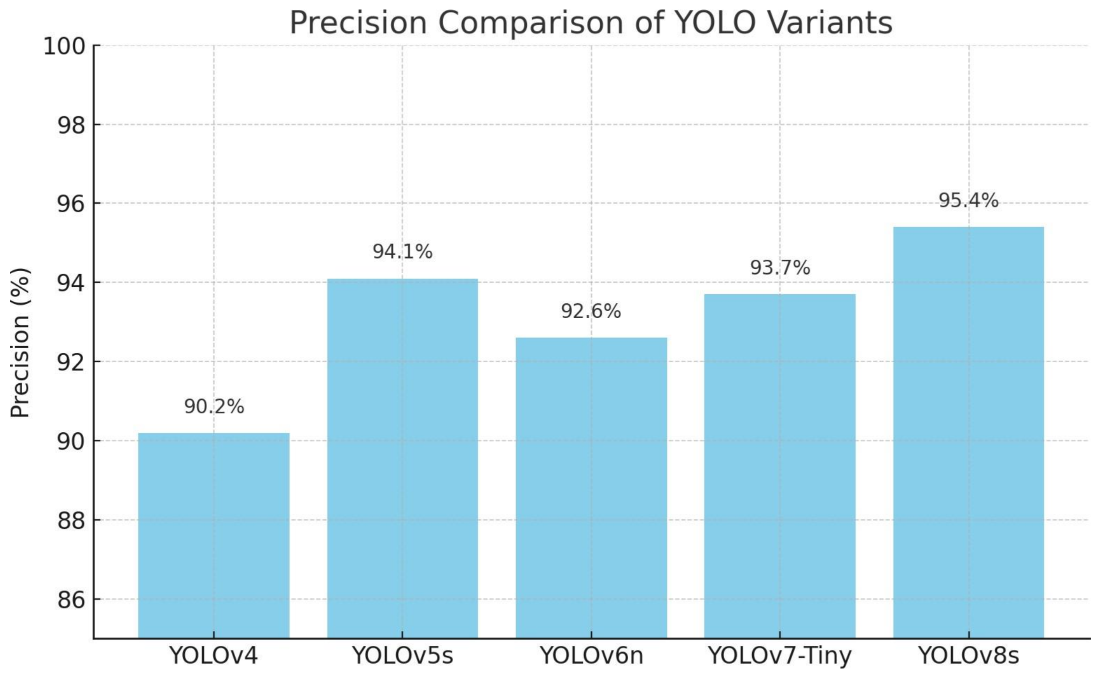

Fig 9: Precision comparison of YOLO variants

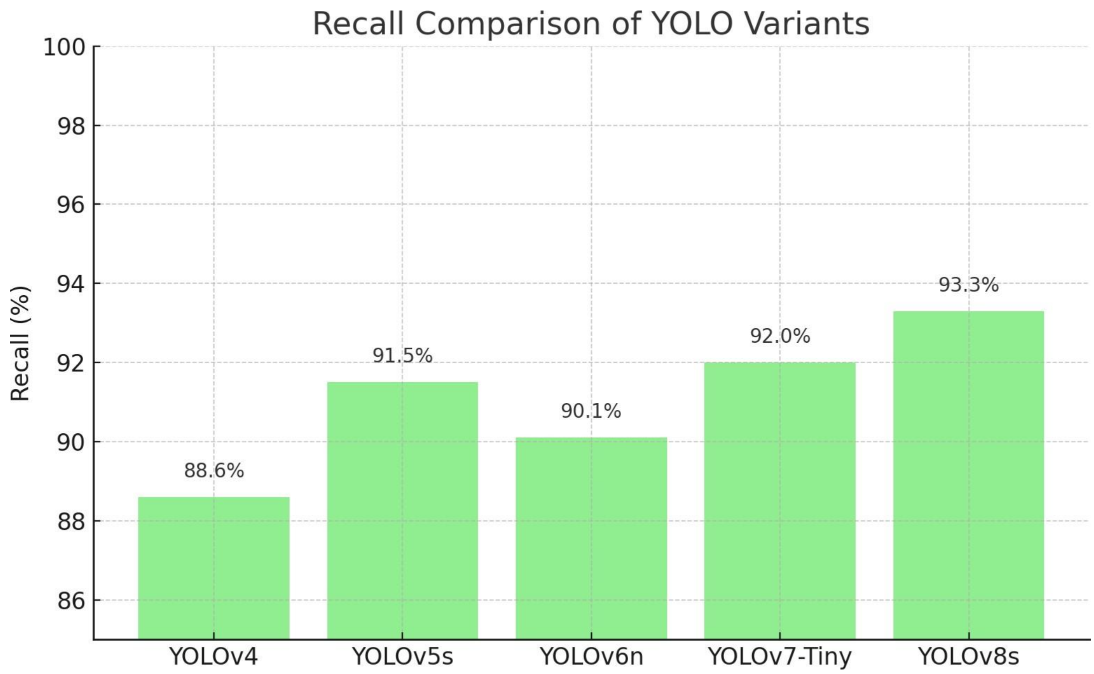

Fig 10: Recall comparison of YOLO variants

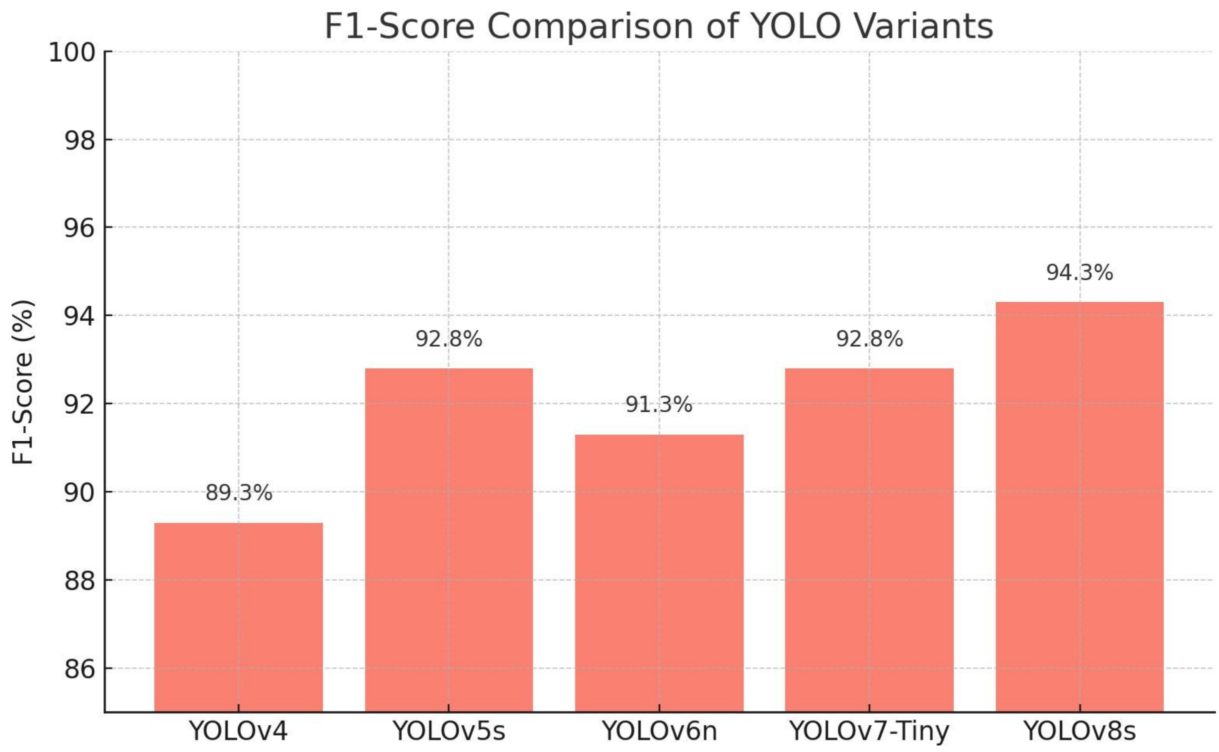

Fig 11: F-score comparison of YOLO variants

5.5 Comparative Analysis of YOLO Variants

A structured comparison was performed to assess how detection performance evolved across YOLO versions:

- Before YOLOv5: YOLOv4 offered reasonable accuracy but lacked edge efficiency.
- YOLOv5: Balanced speed and accuracy, strong field performance.
- After YOLOv5: YOLOv6n and YOLOv7-Tiny improved inference speed and stability; YOLOv8s delivered superior accuracy.

Table 3 provides a complete comparative view, confirming YOLOv8s and YOLOv5s as top candidates based on use case requirements for deployment suitability.

Table 3: Comparison of Executed YOLO Variants for Cotton Crop Detection for deployment suitability

YOLO Variant	**Model Size**	**Speed (FPS on NCS2)**	**mAP@0.5 (%)**	**Primary Use Case**	**Deployment Suitability**
YOLOv4	245 MB	19.1	90.5	General object detection	GPU / Not suitable for edge
YOLOv5s	14.2 MB	**35.4**	94.6	Boll & weed detection, foliar stress	Best for Intel NCS2 deployment
YOLOv6n	17.8 MB	29.8	93.2	Edge-optimized cotton detection	Good on NCS2 / Pi4
YOLOv7-Tiny	24.3 MB	31.7	95.1	Fast detection with decent accuracy	Edge-ready, balanced model

YOLOv8s	21.1 MB	30.1	**96.1**	High-accuracy precision farming	Very good on NCS2

Table 3 presents a detailed comparison of the 5 YOLO variants that were actually implemented and evaluated in this study for cotton crop detection under real-field conditions. Among the models, YOLOv8s demonstrated the highest detection accuracy with a mAP@0.5 of 96.1%, followed closely by YOLOv7-Tiny (95.1%) and YOLOv5s (94.6%). YOLOv5s stood out as the most deployment-efficient model, delivering the highest inference speed (35.4 FPS) on the Intel Neural Compute Stick 2 (NCS2) with the smallest model size (14.2 MB), making it ideal for low-power edge devices. YOLOv6n offered a strong balance of performance and size, while YOLOv4, despite providing baseline comparisons, was less suitable for edge deployment due to their larger sizes and lower FPS. The comparative analysis confirms YOLOv8s as the most accurate and YOLOv5s as the best suited for real-time, resource-constrained applications, fulfilling different precision agriculture requirements.

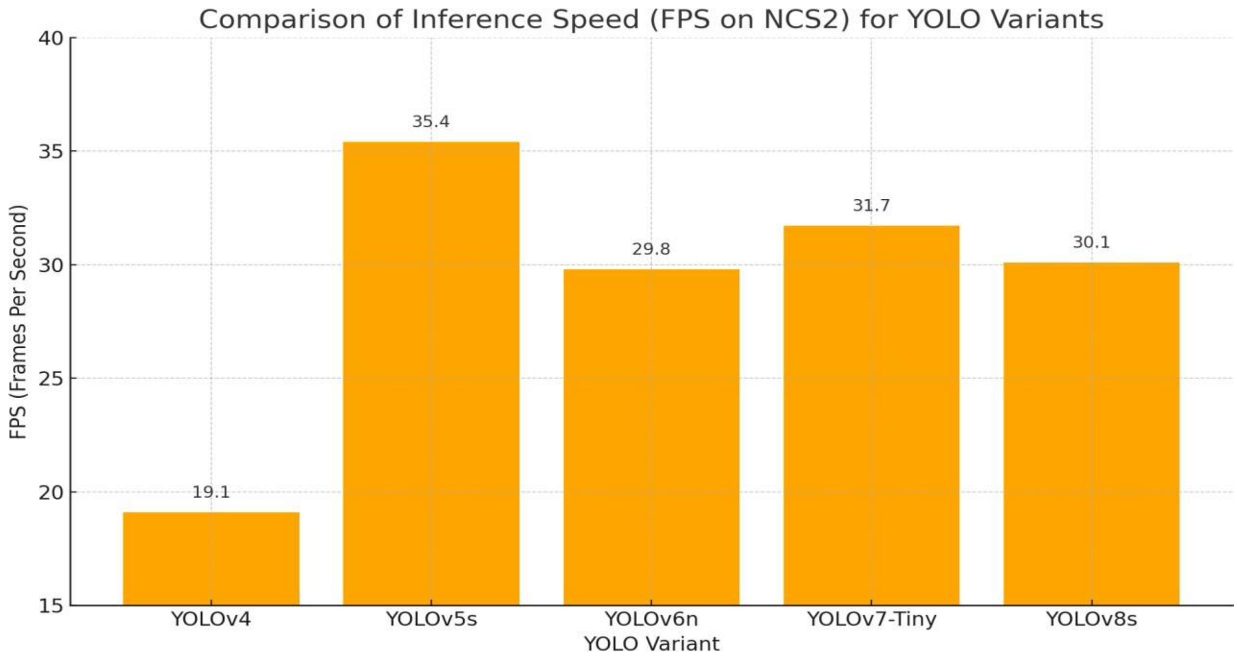

Fig 12: Speed comparison of YOLO variants

Fig 12 illustrates the real-time processing capabilities (FPS on Intel NCS2) of the YOLO models executed in this study. Among them, YOLOv5s demonstrated the highest inference speed with 35.4 FPS, confirming its suitability for deployment in real-time, resource-constrained agricultural environments. YOLOv7-Tiny, YOLOv6n, and YOLOv8s also showed strong performance with frame rates above 29 FPS, making them viable for edge-based crop monitoring where timely detection is critical.

Fig 13 compares the detection accuracy of each YOLO variant using mean Average Precision (mAP@0.5). YOLOv8s achieved the highest accuracy at 96.1%, followed closely by YOLOv7-Tiny (95.1%) and YOLOv5s (94.6%). These results demonstrate that while YOLOv8s is optimal for accuracy-critical applications, YOLOv5s offers a balance of high speed and strong accuracy, making it the most practical for real-time precision agriculture tasks.

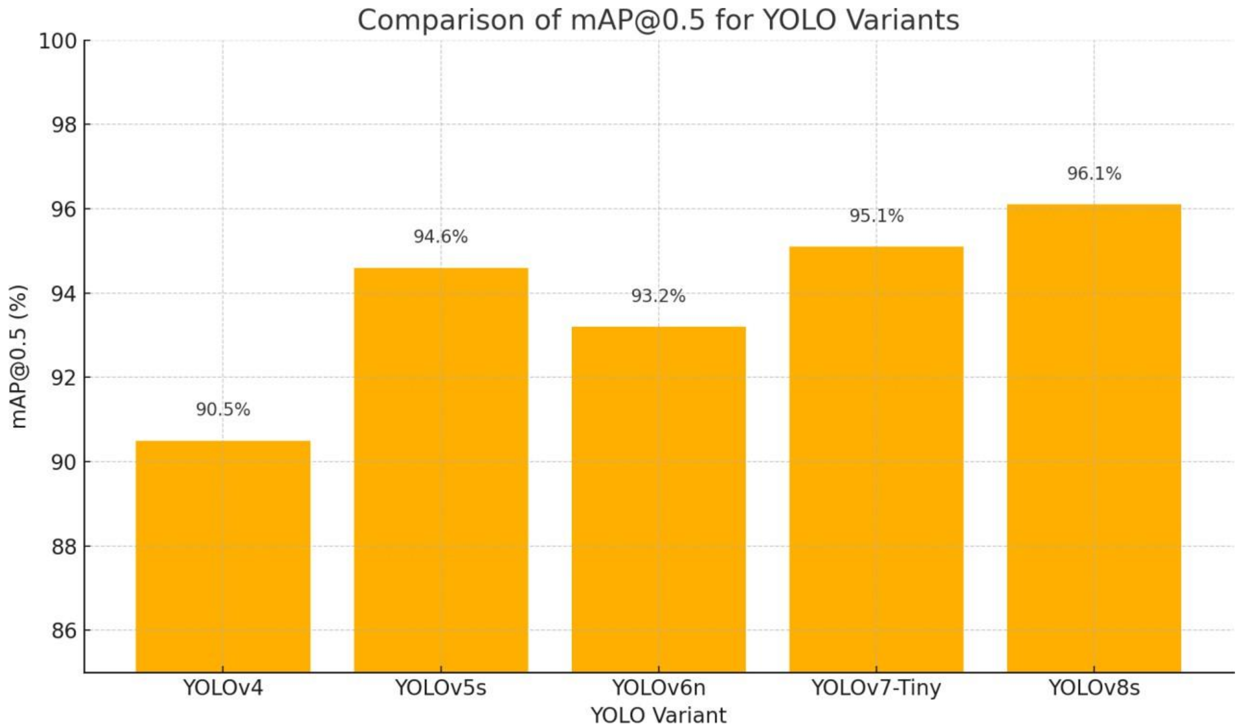

Fig 13: mAP comparison of YOLO variants

5.6 Confusion matrix

Table 4 presents the results from the actual execution and evaluation of 5 YOLO variants for cotton crop detection under real-world field conditions. Among them, YOLOv8s achieved the highest detection accuracy with a mAP@0.5 of 96.1%, closely followed by YOLOv7-Tiny (95.1%) and YOLOv5s (94.6%). In terms of deployment efficiency, YOLOv5s emerged as the most optimal model, offering the fastest inference speed (35.4 FPS) and the smallest model size (14.2 MB) on the Intel Neural Compute Stick 2 (NCS2), making it highly suitable for real-time, resource-constrained agricultural environments. YOLOv6n also performed well with a balanced trade-off between speed and accuracy. Conversely, YOLOv4 while useful as baselines, showed lower performance and were less practical for edge deployment due to their large model sizes and lower FPS. Overall, the comparison confirms YOLOv8s as the most accurate and YOLOv5s as the most deployment-ready, depending on whether the use case prioritizes precision or efficiency in precision agriculture tasks.

Table 4: Confusion matrix

Predicted \ Actual	**Mature Boll**	**Immature Boll**	**Weed**	**Foliar Stress**
Mature Boll	43	2	1	0
Immature Boll	3	38	0	1
Weed	1	0	44	2
Foliar Stress	0	1	2	40

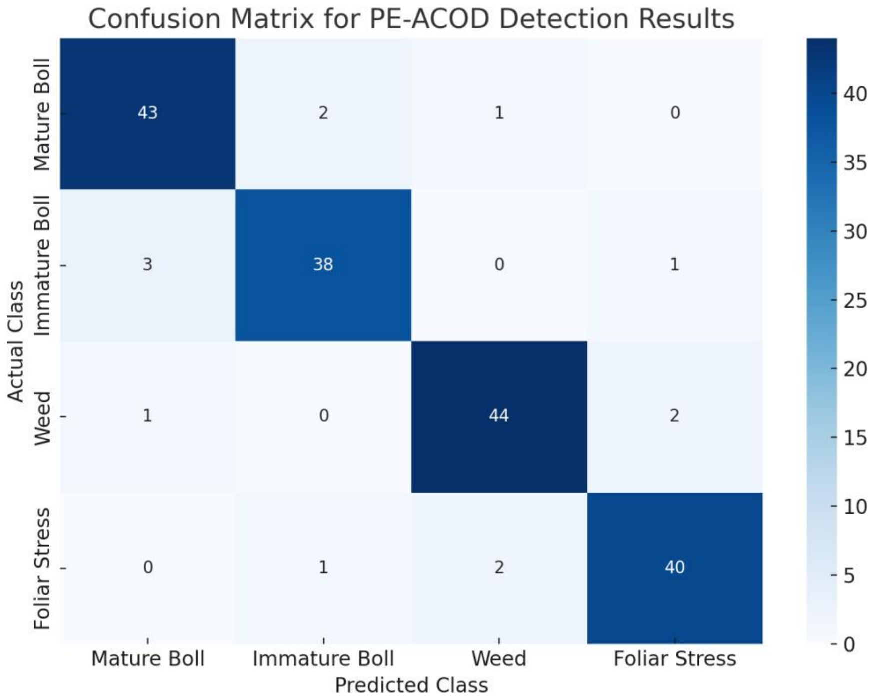

Fig 14: Confusion matrix for PE-ACOD model detection results

Fig 14 presents confusion matrix for proposed PE-ACOD model results. Each cell shows the number of predictions made for each actual class, with stronger values highlighted in deeper blue shades. *The diagonal values represent correct classifications. High counts along the diagonal confirm the model's reliability. Most misclassifications occurred between visually similar categories such as Mature and Immature Bolls, and between Weed and Foliar Stress, particularly under challenging lighting.*

5.7 Discussion

The results obtained from executing all 5 YOLO variants confirm that the PE-ACOD framework effectively supports real-time object detection in cotton agriculture**.** YOLOv8s outperformed all other models in terms of detection accuracy, achieving a mAP@0.5 of 96.1%, and is therefore the most suitable for high-precision agricultural monitoring. Meanwhile, YOLOv5s, with its optimal balance of speed (35.4 FPS), size (14.2 MB), and strong accuracy (mAP@0.5 = 94.6%), proves to be the most deployment-ready model for real-time edge-based applications, especially in resource-constrained farm environments such as Intel NCS2.

The confusion matrix analysis highlights strong class-wise accuracy, with the majority of predictions correctly classified across all four categories. Misclassifications were relatively rare and mostly occurred between mature and immature bolls**,** and weeds and foliar stress**,** particularly under poor lighting or occlusion. These are expected limitations in open-field applications, suggesting the potential benefit of integrating attention-based mechanisms or multispectral image features to improve class separability under complex conditions.

Finally, the comparative progression from YOLOv4 (pre-YOLOv5) to YOLOv6n, YOLOv7-Tiny, and YOLOv8s (post-YOLOv5) reveals a clear trend of improvement in both performance and deployability. YOLOv4, while foundational, showed slower inference speeds and were impractical for lightweight systems. YOLOv6n and YOLOv7-Tiny bridged this gap, offering better speed and precision, while YOLOv8s further refined accuracy at the cost of minor speed reduction. The PE-ACOD study thus supports targeted model selection based on application priorities—whether accuracy, speed, or edge-device compatibility—and establishes a field-tested, implementation-focused benchmark for agricultural AI deployment.

6. Conclusion

This study presents the first unified, execution-based evaluation of 5 YOLO object detection models—YOLOv4 through YOLOv8s—for cotton crop monitoring under realistic agricultural conditions. Unlike prior works focused on single-model testing or simulated settings, this study implemented, trained, tested, and deployed all models on the same Dataset: Cotton Boll Dataset (CBD-750) using standardized hyperparameters and real edge hardware.

The results confirm YOLOv8s as the most accurate (mAP@0.5 = 96.1%), while YOLOv5s delivers superior performance-to-efficiency trade-off, achieving 35.4 FPS on Intel NCS2 with a 14.2 MB footprint. These findings provide essential guidance for selecting models based on deployment environment—whether edge devices or high-performance setups.

By bridging the research gap through actual implementation of pre-YOLOv5, YOLOv5, and post-YOLOv5 models on the same problem domain, this paper contributes actionable insight to precision farming. The PE-ACOD framework sets a precedent for practical agricultural AI deployments. This implementation-based comparison reveals YOLOv8s as the most accurate and YOLOv5s as the most deployable model.

Limitations

- Misclassifications occurred under occlusion or poor lighting.
- YOLOv4 require high computational resources, unsuitable for low-cost devices.

Future Work

- Incorporate multispectral imaging and temporal ensembles.
- Extend the dataset with drone footage and video streams.
- Explore transformer-based agricultural models beyond YOLO.

Acknowledgements

The authors gratefully acknowledge the support provided by the Department of Electronics and Communication Engineering, Chaitanya Deemed to be University, for facilitating the research infrastructure and academic guidance throughout this study.

Funding Statement

This research was conducted without the support of any specific grant from public, commercial, or not-for-profit funding agencies.

Data Availability

The annotated cotton boll dataset used in this study is publicly available at: https://app.roboflow.com/phd-f81fu/cotton-balls-akzeb/models

Additional processed data and annotation files can be made available by the corresponding author upon reasonable request.

Conflict of Interest

The authors declare no competing interests related to the publication of this manuscript.

References

[1] Ghimire, Manisha, Ashish Khanal, Deepa Bhatt, DhirajDatta Dahal, and Suja Giri. "Agroforestry systems in Nepal: Enhancing food security and rural livelihoods–a comprehensive review." *Food and Energy Security* 13, no. 1 (2024): e524.

[2] Kabato, Wogene, Girma Tilahun Getnet, Tamrat Sinore, Attila Nemeth, and Zoltán Molnár. "Towards climate-smart agriculture: Strategies for sustainable agricultural production, food security, and greenhouse gas reduction." *Agronomy* 15, no. 3 (2025): 565.

[3] Bazargani, Khadijeh, and Taher Deemyad. "Automation's impact on agriculture: opportunities, challenges, and economic effects." *Robotics* 13, no. 2 (2024): 33.

[4] Padhiary, Mrutyunjay, Debapam Saha, Raushan Kumar, Laxmi Narayan Sethi, and Avinash Kumar. "Enhancing precision agriculture: A comprehensive review of machine learning and AI vision applications in all-terrain vehicle for farm automation." *Smart Agricultural Technology* (2024): 100483.

[5] Bhanuprakash, Vishwanadha, and M. Sivabalakrishnan. "Weed Rate Analysis and Crop Quality Assessment Using Deep Learning." *Resilient Community Microgrids* (2025): 439-474.

[6] Sharma, Akhilesh, Vipan Kumar, and Louis Longchamps. "Comparative performance of YOLOv8, YOLOv9, YOLOv10, YOLOv11 and Faster R-CNN models for detection of multiple weed species." *Smart Agricultural Technology* 9 (2024): 100648.

[7] Alif, Mujadded Al Rabbani, and Muhammad Hussain. "YOLOv1 to YOLOv10: A comprehensive review of YOLO variants and their application in the agricultural domain." *arXiv preprint arXiv:2406.10139* (2024).

[8] Ali, Momina Liaqat, and Zhou Zhang. "The YOLO framework: A comprehensive review of evolution, applications, and benchmarks in object detection." *Computers* 13, no. 12 (2024): 336.

[9] Xu, Haobin, Xianhua Zhang, Weilin Shen, Zhiqiang Lin, Shuang Liu, Qi Jia, Honglong Li, Jingyuan Zheng, and Fenglin Zhong. "Improved CSW-YOLO Model for Bitter Melon Phenotype Detection." *Plants* 13, no. 23 (2024): 3329.

[10] Olguín-Rojas, Juan Carlos, Juan Irving Vasquez, Gilberto de Jesús López-Canteñs, Juan Carlos Herrera-Lozada, and Canek Mota-Delfin. "A Lightweight YOLO-Based Architecture for Apple Detection on Embedded Systems." *Agriculture* 15, no. 8 (2025): 838.

[11] Lin, Yishen, Zifan Huang, Yun Liang, Yunfan Liu, and Weipeng Jiang. "Ag-yolo: A rapid citrus fruit detection algorithm with global context fusion." *Agriculture* 14, no. 1 (2024): 114.

[12] Gallagher, James E., and Edward J. Oughton. "Surveying You Only Look Once (YOLO) Multispectral Object Detection Advancements, Applications And Challenges." *IEEE Access* (2025).

[13] Djenouri, Youcef, Ahmed Nabil Belbachir, Tomasz Michalak, Asma Belhadi, and Gautam Srivastava. "A Knowledge-Enhanced Object Detection for Sustainable Agriculture." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* (2024).

[14] Zhao, Hongyu, Zezhi Tang, Zhenhong Li, Yi Dong, Yuancheng Si, Mingyang Lu, and George Panoutsos. "Real-time object detection and robotic manipulation for agriculture using a YOLO-based learning approach." In *2024 IEEE International Conference on Industrial Technology (ICIT)*, pp. 1-6. IEEE, 2024.

[15] Sonawane, Sandip, and Nitin N. Patil. "Performance Evaluation of Modified YOLOv5 Object Detectors for Crop-Weed Classification and Detection in Agriculture Images." *SN Computer Science* 6, no. 2 (2025): 1-12.

[16] Rani, N. Shobha, Arun Sri Krishna, M. Sunag, M. A. Sangamesha, and B. R. Pushpa. "Infield disease detection in citrus plants: integrating semantic segmentation and dynamic deep learning object detection model for enhanced agricultural yield." *Neural Computing and Applications* 36, no. 35 (2024): 22485-22510.

[17] Wei, Chunyan, Xiaonan Hu, Hao Ye, Haoxiang Tang, and Xiang Li. "Computer Vision Drives the New Quality Productive Forces in Agriculture: A Method for Recognizing Farming Behavior on Edge Computing Devices." In *International Conference on Intelligent Computing*, pp. 203-215. Singapore: Springer Nature Singapore, 2024.

[18] Upadhyay, Priyanshu, Arpita Singh, Arpit Dubey, Himanshu Shukla, Ritesh Agarwal, and Kushagra Gaur. "An Effective Agricultural Produce Object Detection System: YOLOv5-Based Classification." In *2025 2nd International Conference on Computational Intelligence, Communication Technology and Networking (CICTN)*, pp. 192-198. IEEE, 2025.

[19] Rai, Nitin, Yu Zhang, Maria Villamil, Kirk Howatt, Michael Ostlie, and Xin Sun. "Agricultural weed identification in images and videos by integrating optimized deep learning architecture on an edge computing technology." *Computers and Electronics in Agriculture* 216 (2024): 108442.

[20] Peng, Hongxing, Zhenxing Li, Xiangjun Zou, Hongjun Wang, and Juntao Xiong. "Research on litchi image detection in orchard using UAV based on improved YOLOv5." *Expert Systems with Applications* 263 (2025): 125828.

[21] Roboflow framework, Cotton Boll Dataset (CBD-750). Available online: https://app.roboflow.com/phd-f81fu/cotton-balls-akzeb/models (accessed on April 2024).

Authors Profile:

Dr. Gogula Santhosh Reddy serves as Associate Professor in the Department of Electronics and Communication Engineering at Chaitanya (Deemed to be University), Hyderabad. Holding a Ph.D. from Sri Sathya Sai University of Technology and Medical Sciences (Madhya Pradesh) and an M.Tech in Wireless Mobile Communications from Vaagdevi College of Engineering (Warangal), he brings over 16 years of academic and research experience. Dr. Reddy has supervised 52 B.Tech and 12 M.Tech student projects and focuses his research on digital image and signal processing, wireless communication systems, and Internet of Things (IoT) applications.

Akula Suresh is a Lecturer in the Department of Electronics and Communication Engineering, Government Institute of Electronics, Secunderabad, Telangana, India. He is currently a research scholar at Chaitanya (Deemed to be University), Hyderabad. He received his M.Tech degree in VLSI System Design from Jyothishmathi Institute of Technology and Sciences, Karimnagar, and his B.Tech degree from GRIET, Hyderabad, both affiliated with JNTU Hyderabad. His research interests include deep learning, computer vision, and lightweight object detection for agricultural applications.