

REAL-TIME DATE RIPENESS CLASSIFICATION USING YOLOV11 FOR AGRICULTURAL AUTOMATION

Aqeel Ahmed^{*1}, Kamran Dahri², Faheem Ahmed Abbasi³, Altaf Mazhar Soomro⁴,
Mudasir Ahmed Memon⁵

^{*1}Dr. A.H.S Bukhari Post Graduate Centre of ICT, University of Sindh, Jamshoro Pakistan,

^{2,3}Department of Information Technology, University of Sindh, Jamshoro Pakistan,

⁴University of Technology, Sydney Australia

⁵Department of Electronics Engineering, University of Sindh

^{*1}dotaqeel@gmail.com, ²kamran.dahri@usindh.edu.pk, ³faheem.abbasi@usindh.edu.pk,

⁴altaf.m.soomro@alumini.uts.edu, ⁵memon.mudasir@usindh.edu.pk

DOI: <https://doi.org/10.5281/zenodo.22201157>

Keywords

YOLO (You Only Look Once),
Ripen AI, Object Detection,
Deep Learning, Date Fruits.

Article History

Received: 02 April 2026

Accepted: 12 May 2026

Published: 31 May 2026

Copyright @Author

Corresponding Author: *

Aqeel Ahmed

Abstract

Conventional methods of evaluating the maturity of date fruits are usually heavy on the workers, depend a lot on the judgment of the person, and are quite slow, especially in big-scale agricultural setups where speed and uniformity are of great importance. To get over these drawbacks, this study is introducing Ripen AI, a highly sophisticated automated system that can amazingly classify the maturity of the date fruits within a very short time and with an impeccable accuracy rate. The system makes use of YOLOv11, which is the newest type of single-stage very deep learning object detector, and it considers object detection a single regression task. The method of object detection can be done without the need for classification at all which is very fast but still maintains high classification accuracy. Ripen AI has specifically been designed to locate and classify date fruits that belong to one of the four main groups of maturation stages, which are Immature, Khalal, Rutab, and Tamar. In addition, it can handle the cases where the lights are varied, occlusions occur, as well as the exterior environmental factors are changing. A device employing YOLOv11 is able to become a real-time model, and therefore, its deployment during field activities, as well as continuous monitoring, is quite a practical idea. With the implementation of Ripen AI, the reliance on manual inspection is lowered, human errors are greatly minimized, and the whole harvesting procedure is greatly sped up. Thus, the correctness and the speed achieved can, in turn, bring about better yield prediction, optimum harvest timing, and, in general, enhancement of the quality of the crop. This kind of system is a clear example of how deep learning can revolutionize the process of agriculture, thus presenting a viable and reliable solution that is in line with modern smart farming applications and, additionally, is scalable.

1. Introduction

Date farming is among the most valuable and environmentally friendly agricultural practices

that spread widely across different counties of the world, mainly in dry and semi-dry climates. The impressive lifespan of a single date palm tree is

from 40 to 50 years, during which farmers get economic stability and a regular source of income [1]. However, the most significant factor limiting the quality of the yield, the market price, and the readiness for export is the correct determination of the fruit maturity level [2][3]. In the case of the visual examination and personal experience, it is the traditional method that farmers use to decide the right moment for the harvest. Although the manual method is functional, it is subject to bias, takes a lot of time, and is susceptible to human mistakes [4][5]. It happens because light and other conditions such as fruit color, cluster density, and environment may affect viewing and thus result in different judgments on maturity, making the picking of fruit either early or late. Such mistakes affect significantly the percentage of fruit going to waste during the post-harvest period, its quality decrease, and the missing of market opportunities [6][7][8].

Ripen AI is a project that started to introduce an innovative, intelligent, and automatic technology to solve these problems. The device uses computer vision and deep learning (YOLOv11) to analyze date fruit maturity accurately. By importing novel breakthroughs in artificial intelligence, in particular, the latest object detection models, the platform aims at meeting farmers' needs for a stable, objective tool that can tell them the maturity of their yields in real-time. The technological upgrade is persuasive in that it lessens mistakes made by humans and thus standardizes outcomes that can be repeated in any farming environment [9][10][11].

The main goal of Ripen AI is to simplify the harvesting process by letting farmers know the exact time when they should collect their fruits. In the long run, the benefits are higher productivity, production of high-quality fruits, and better readiness for the local and international markets. On the top of that, the system allows saving resources, reducing the workforce, and encouraging a data-driven culture, provided farmers delegate the task of maturity classification to the system. To put it differently, Ripen AI aims to become one of the leading components of the smart farming revolution and offer a versatile

solution which, on the one hand, leads to a more efficient date farming operation, and on the other hand, yields better economic outcomes. [7][9][10].

1.1 Problem Statement

Accurate ripeness evaluation of date fruits is critical for guaranteeing product quality, maximizing price and harvesting efficiency. Manual visual inspection, by and large, is dependent on the inspector's skill level, judgment, and experience that renders the outcome subject to variation, laborious, and inconsistent.

1.2 Research Objectives

1. Build a system to automatically detect the ripeness of date fruits.
2. Using YOLOv11m. Classify dates fruits into different Immature, Khalal, Rutab, and finally Tamar stages.
3. Evaluate real-time performance.
4. Deployment through Streamlit.

1.3 Research Questions

- **RQ1:** Can YOLOv11m accurately classify ripeness in real time?
- **RQ2:** Can it reduce manual inspection limitations?

1.4 Significance of the Study

The proposed system helps with agriculture by getting rid of the mistakes that humans make and also gives better harvesting decisions.

2. Research Gap

2.1 Most previous studies:

- Used traditional CNN-based approaches or older YOLO versions.
- Focused primarily on classification rather than real-time object detection.
- Did not provide practical deployment for field use.
- Lacked integration with live monitoring systems and smart farming applications.
- Did not utilize the latest YOLOv11 architecture.

2.2 Proposed work:

- Employs the latest YOLOv11m model.
- Performs real-time object detection and ripeness classification simultaneously.
- Supports both webcam-based and image-based inference.
- Provides a deployable solution through a Streamlit interface.

- Establishes a foundation for future robotic harvesting and precision agriculture systems.

3. Assessment with Current Linked Research

This work shows the results of our study through an advancement in related works given in Table 1 and Table 2 which offers an exact comparison of features between the currently used methods and the proposed system.

Table 1. Comparison of Recent Related Research

Year	Research Work	Method Used	Contribution	Limitations	Difference from Proposed Work
2019	Date Fruit Classification for Robotic Harvesting in a Natural Environment	Deep CNN	Classified date fruit maturity in natural environments	No real-time object detection; older architecture	Uses YOLOv11m for real-time detection and classification
2023	Smart Harvesting Decision System for Date Fruit Based on YOLO and K-Means	YOLO + K-Means	Fruit detection and maturity estimation	Relied on older YOLO models and limited deployment capabilities	Uses the latest YOLOv11 and supports real-time deployment
2023	Date Fruit Detection Dataset for Automatic Harvesting	Dataset Development	Developed a benchmark dataset for harvesting research	No intelligent detection or deployment framework	Provides a complete AI-based detection and classification system
2024	DPXception: A Lightweight CNN for Image-Based Date Palm Classification	CNN (DPXception)	Classified date palm species	Focused on species classification rather than ripeness detection	Performs ripeness-stage detection and localization
2024	YOLOv9-Based Object Detection Studies	YOLOv9	Improved object detection performance	Not applied to date fruit ripeness classification	Applies a newer YOLOv11 model to date maturity detection
2024	YOLOv10-Based Detection Systems	YOLOv10	End-to-end object detection	No application to date fruit maturity analysis	Extends detection to agricultural ripeness classification
2026	Ripen AI (Proposed Work)	YOLOv11m + Streamlit	Real-time detection and classification of	Performance may decrease under severe	First implementation using YOLOv11m with a real-time

			four date fruit maturity stages	occlusion and poor lighting	deployment interface
--	--	--	---------------------------------	-----------------------------	----------------------

Table 2. Comparison of Existing Methods with Proposed Method Features

Study	Year	Method	Real-Time Detection	Deployment Support	Number of Ripeness Stages
Altaheri et al.	2019	Deep CNN	No	No	Multiple
Zarouit et al.	2023	YOLO + K-Means	Partial	Limited	Multiple
YOLOv9 Studies	2024	YOLOv9	Yes	Limited	Not Focused on Dates
Proposed Ripen AI	2026	YOLOv11m	Yes	Yes	4

4. Methodology and Model Architecture

Research Design: This study adopts an experimental quantitative design based on deep learning and computer vision.

4.1 Dataset and Maturity Stages

The Ripen AI project utilized the Date Fruit Dataset for Automated Collecting and Visual Harvest Approximation, a publicly accessible benchmark dataset tailored for agricultural

computer vision tasks. The dataset includes 8,079 high-resolution color images covering over 350 distinct date bunches from 29 palm trees. The imagery spans all key developmental phases are shown in Figure 1:

- **Immature:** Hard, Green Fruits.
- **Khalal:** Yellowish, semi-ripe.
- **Rutab:** Soft, Golden Yellow, Partially Brown.
- **Tamar:** The fruit is fully ripe, Red, and Dark Brown.

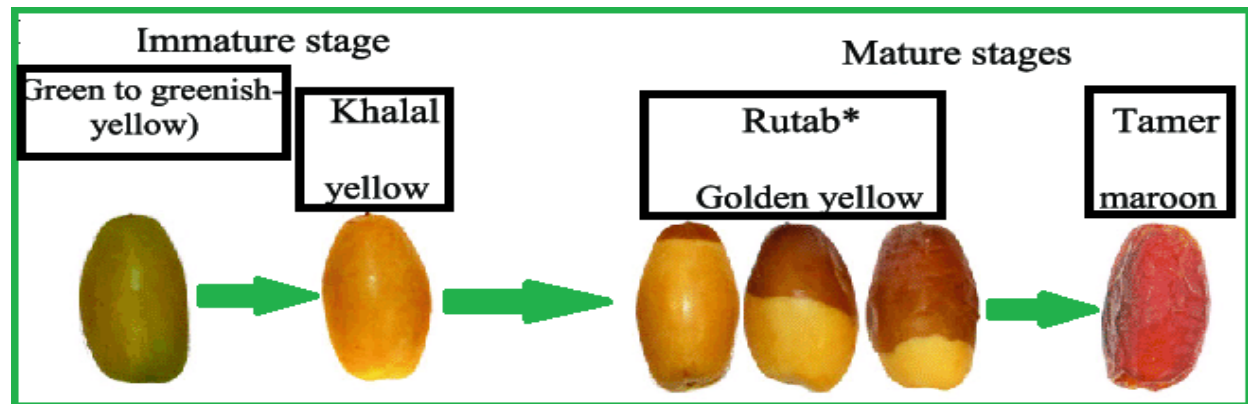


Figure 1. Some sample photos of date fruit at different stages of the maturity from the dataset.

The imagery spans all key developmental phases as detailed in Table 3 and Table 4:

Table 3. Ripeness Categories Used in This Study

Class ID	Ripeness Stage	Description
0	Immature	Hard and green fruit
1	Khalal	Yellowish and semi-ripe fruit

2	Rutab	Soft golden-yellow and partially brown fruit
3	Tamar	Fully ripe dark brown fruit

Table 4. Dataset Distribution Across Ripeness Stages

Ripeness Stage	Number of Images	Percentage
Immature	2,020	25.0%
Khalal	2,020	25.0%
Rutab	2,019	25.0%
Tamar	2,020	25.0%
Total	8,079	100%
<i>(Note: The dataset classes were balanced during preprocessing and augmentation.)</i>		

(Refer to Figure 1 for sample photos of date fruit at different stages of maturity from the dataset).

The dataset was made strong by including different environmental conditions like direct sunlight, shadows, bad lighting, shots from different angles, and occlusions (where one part was hidden, for example under, protective bags).

4.2 Model Architecture (YOLOv11)

We have decided to go with the YOLOv11m ("medium") model here. This model is the newest from Ultralytics which is why it has the fastest performance and multi-tasking efficiency. It is the essence of YOLO that it treats object detection as one single, unified task and in one image scan can predict the localization and classification of the object simultaneously. As a feature such approach gives very high speed [8][11][12]. Details about the changes in architecture that went into developing YOLOv11 can be found at Table 5.

The study used several standard evaluation metrics to evaluate the date ripeness classification performance of the YOLOv11m model. Below is the formula for each of those evaluation metrics:

- **Precision (P):** Indicates how many selected items were correctly identified.

$$P = \frac{TP}{TP + FP}$$

(Where TP = True Positives, and FP = False Positives)

- **Recall (R):** This metric assesses how well the model retrieves the full set of relevant cases.

$$R = \frac{TP}{TP + FN}$$

(Where FN = False Negatives)

- **F1-Score (F₁):** The geometric average calculated from the Precision and Recall metrics, resulting in an index that represents the compromise between Precision and Recall.

$$F1 = 2 * \frac{P * R}{P + R}$$

- **Frames Per Second (FPS):** It measures how many images per second a machine or system can process and show without lagging. It's closely related to the processing time per image.

$$FPS = \frac{1}{\text{Inference Time (in Seconds)}}$$

Table 5. Evolution of YOLO Architectures and Key Innovations

Model	Year	Key Innovations and Enhancements
YOLOv1	2015	Divided the image into a 7×7 grid and predicted bounding boxes and class probabilities in a single forward pass.

YOLOv2	2016	Improved architecture with batch normalization, anchor boxes, and dimension clustering.
YOLOv3	2018	Improved detection performance via a more efficient backbone network and multi-scale predictions.
YOLOv4	2020	Implemented Mosaic data augmentation, a better detection head, and changes in loss functions.
YOLOv5	2020	Focused on performance optimization and usability, adding automated hyperparameter tuning.
YOLOv6	2022	Open-source release by Meituan used in autonomous delivery systems.
YOLOv7	2022	Proposed Efficient Layer Aggregation Networks (E-ELAN).
YOLOv8	2023	Greater adaptability for an immense variety of computer vision applications.
YOLOv9	2024	Introduced Programmable Gradient Information (PGI) and Generalized Efficient Layer Aggregation Network (GELAN).
YOLOv10	2024	Developed a new end-to-end detection framework eliminating Non-Maximum Suppression (NMS).
YOLOv11	2024	The most recent Ultralytics update, setting new records for detection, segmentation, and classification.

Key advantages of the YOLOv11 architecture are:

1. **SOTA Performance:** It provides such a high performance (i.e., a precision level of a very high one) which is a crucial factor for identifying the small fruits that are often occluded and hard to detect especially in date palm areas.
2. **Task Flexibility:** It is a framework that is so adaptable that besides being able to run object detection, you can also use it for segmentation.
3. **Innovations for Efficiencies:** A head that goes all the way from a prediction to the final output which can save time as we won't have to

NMS post-processing. Besides that, the model uses C3K2 Blocks and C2PSA Attention to bring more complex features to the level of detail.

4.3 Data Preparation and Augmentation

First, we uploaded the pictures in Roboflow to make annotations. Then each fruit was circled with a tight bounding box, and assigned one ripeness of four labels as per Figure 2. Afterwards, Roboflow converted all those bounding boxes into YOLO TXT [13][14] format and randomly split the dataset following the scheme shown in Table 6.

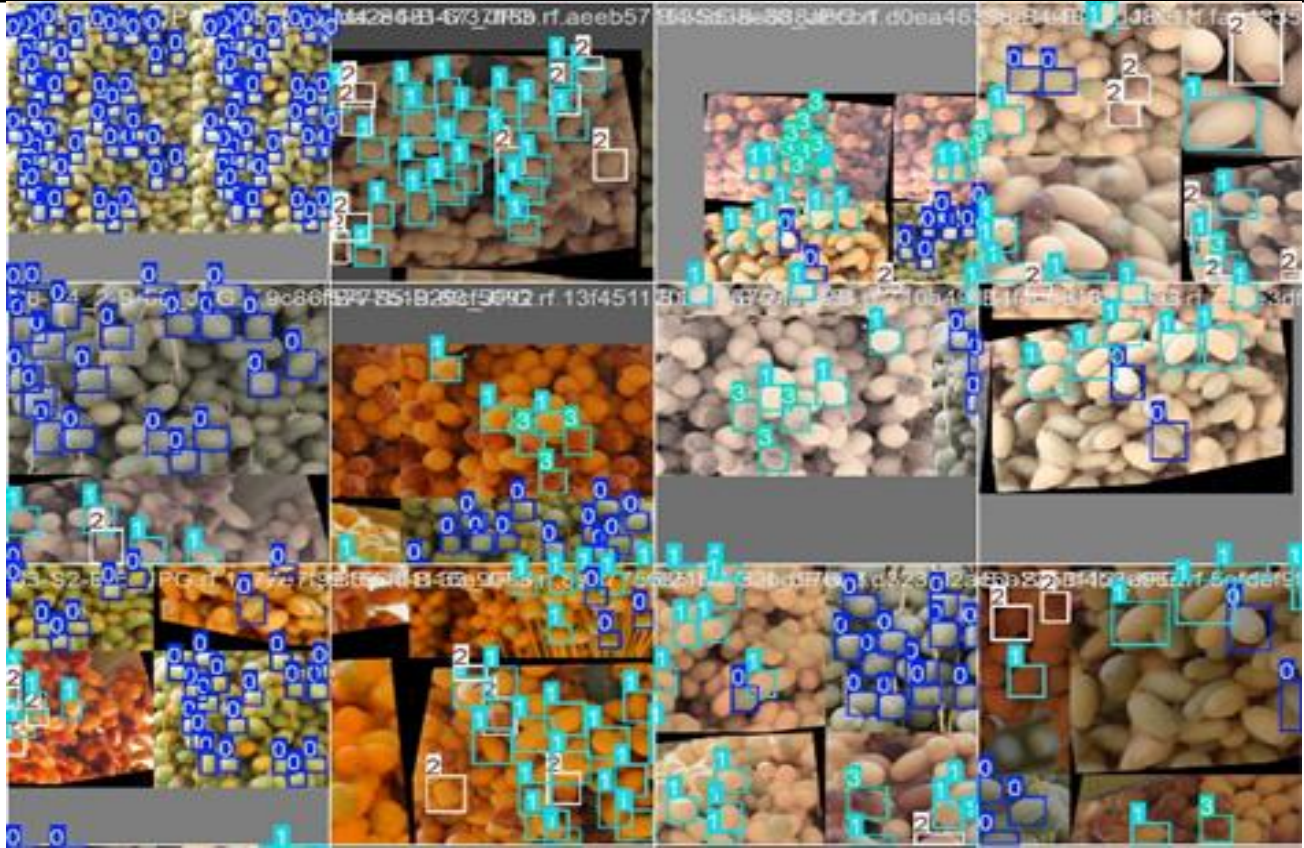


Figure 2 Data Set Labeling Boxing

The research reveals that the entire dataset includes 8, 079 images. Naturally, the data is distributed as follows: 80% for training purpose, 10% for validation, and 10% for testing.

This is illustrated mathematically below.

- **Training Set:** $8079 * 0.80 = 6463.2$ (Rounded to 6,463 images).

- **Validation Set:** $8079 * 0.10 = 807.9$ (Rounded to 808 images).
- **Testing Set:** $8079 * 0.10 = 807.9$ (Rounded to 808 images).
- **Verification:** $6463 + 808 + 808 = 8079$ total images. This matches Table 6 perfectly.

Table 6. Dataset Split Configuration

Dataset Partition	Number of Images	Percentage
Training Set	6,463	80%
Validation Set	808	10%
Testing Set	808	10%
Total	8,079	100%

(Refer to Figure 2 for Data Set Labeling Boxing).

Data augmentation, or the addition and transformation of available samples, allowed us to enrich our data diversity significantly. All our methods are presented in Table 7 with references.

Table 7. Data Augmentation Techniques Used

Augmentation Technique	Parameters	Purpose
Auto-Orient	EXIF Correction	Standardize image orientation (Figure 3)
Resize	640 × 640 pixels	Match YOLOv11 input size with letterbox padding (Figure 4)
Horizontal & Vertical Flip	Random	Increase spatial diversity and generalization (Figure 5)
Noise Injection	Random Level	Improve robustness in low-light conditions (Figure 6)
Gaussian Blur	Random Kernel	Simulate blurry/out-of-focus images (Figure 7)
Rotation	$\pm 15^\circ$	Handle camera angle variations (Figure 8)
Shear Transformation	$\pm 10^\circ$	Simulate perspective distortions (Figure 9)

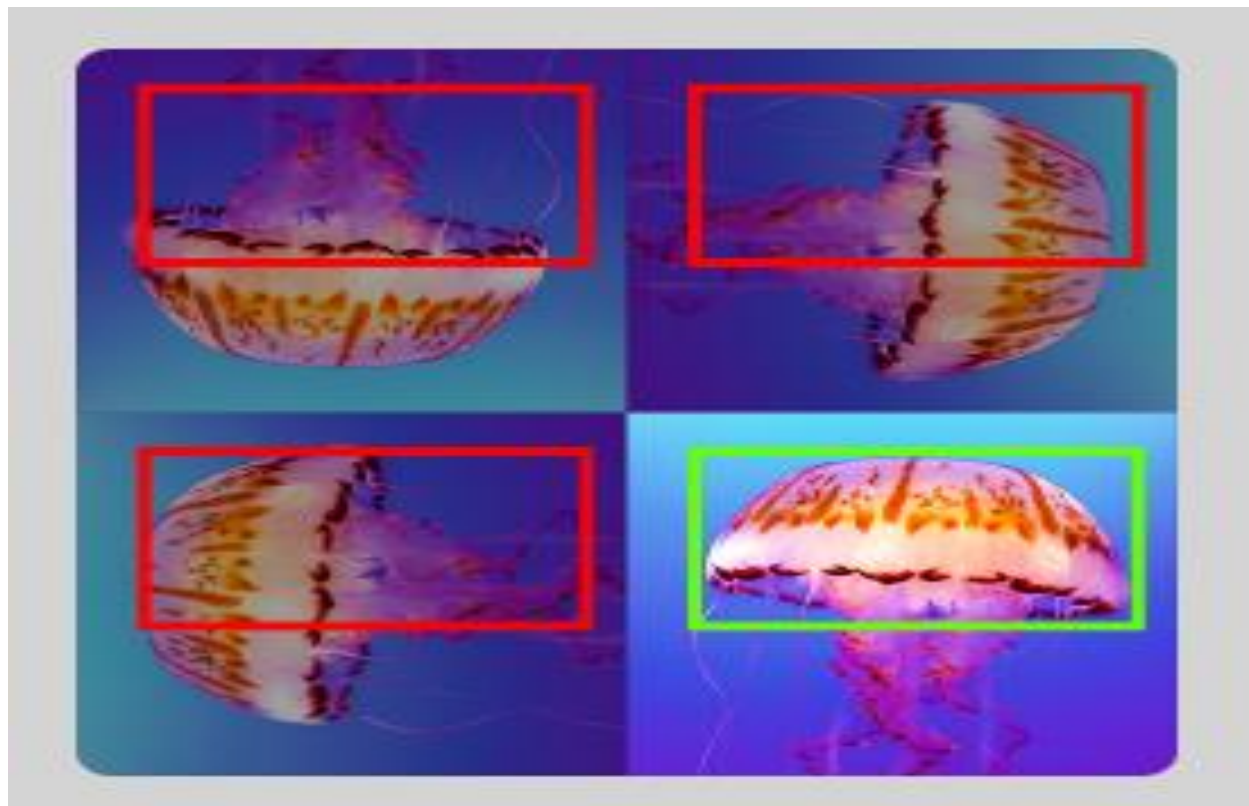


Figure 3. Auto-Orient



4.4 Hardware, Software, and Training Configuration

In the interests of replication, Table 8 and Table 9 contain the complete information of the model's

environmental setup and training arrangements, respectively.

Table 8. Hardware and Software Environment

Component	Specification
Operating System	Windows 11
Programming Language	Python 3.x
Deep Learning Framework	PyTorch
Detection Framework	Ultralytics YOLOv11
Annotation Platform	Roboflow

Deployment Framework	Streamlit
Image Processing Lib	OpenCV
Model Format	PyTorch (.pt)

Table 9. Training Configuration of YOLOv11m Model

Parameter	Value
Model	YOLOv11m
Epochs	40
Input Image Size	640 × 640
Number of Classes	4

5. Deployment and Results

The prototype was trained for a total of 40 epochs. From the results it looks like the loss kept declining during training. Refer to Table 10 and Table 11, the model got excellent global and per-class performance metrics. The fact that

(mAP) is a large mean and inference time is only 18 milliseconds per image and 55 frames per second demonstrates that the model is continuously learning and has a great potential for real-time application as illustrated in Figure 10.

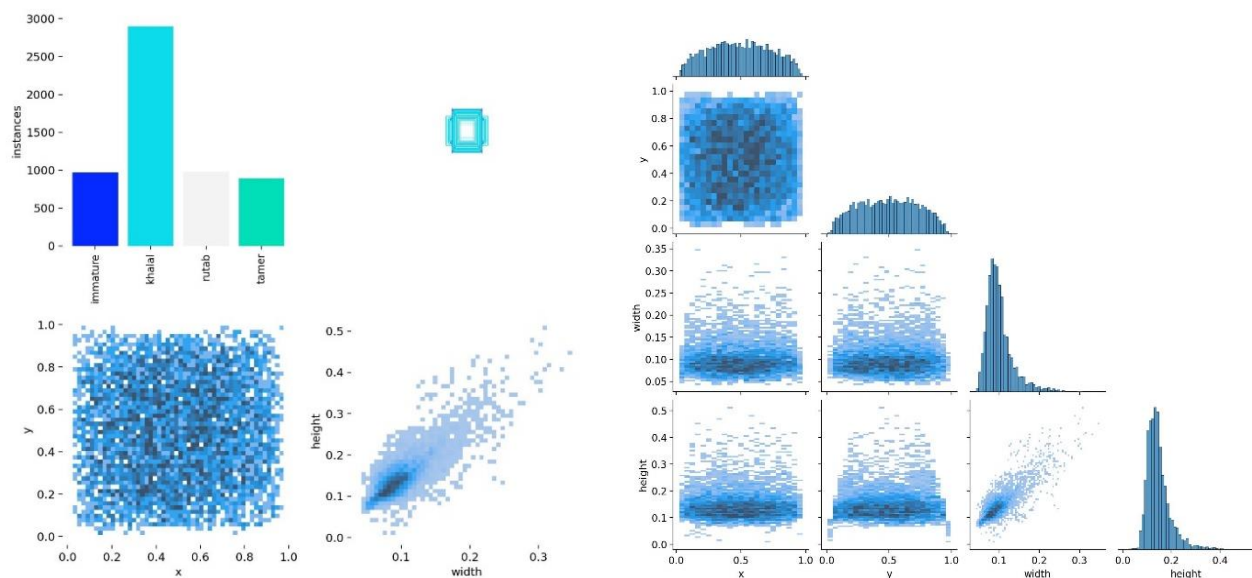


Figure 10. Training loss and mean average bar graph

Table 10 reports the overall performance metrics for the YOLOv11m model:

- **Precision (P):** 95.8% or 0.958.
- **Recall (R):** 94.6% or 0.946.
- **Reported F1-Score:** 95.2%.

Using the F1-Score formula, we can verify this calculation:

$$F1 = 2 * \frac{0.958 * 0.946}{0.958 + 0.946}$$

$$F1 = 2 * \frac{0.906268}{1.904}$$

$$F1 = 2 * 0.475981$$

$$F1 \approx 0.95196$$

- **Verification:** we simply do the conversion; we get that 0.95196 is equivalent to roughly 95.2 %. And there we have the exact value of the overall F1-Score being referred to in Table 10. This is, therefore, a very good check for the data.

Table 10. Overall Performance Metrics

Metric	Value
Precision	95.8%
Recall	94.6%
F1-Score	95.2%
mAP@0.5	96.7%
mAP@0.5:0.95	91.8%
Inference Time	18 ms/image
FPS	55

The study claims the YOLOv11m model is capable of real-time processing, reporting an Inference Time of **18 ms/image** and an FPS of **55**.

To mathematically calculate the FPS from the inference time:

1. Convert milliseconds to seconds: 18 ms = 0.018 seconds

2. Apply the FPS formula:

$$FPS = \frac{1}{0.018}$$

$$FPS \approx 55.55$$

- **Verification:** The calculated value of **55.55** closely aligns with the reported **55 FPS** in Table 10 (likely rounded down to represent guaranteed sustainable frames per second).

Table 11. Class-wise Performance Evaluation

Class	Precision	Recall	F1-Score	mAP@0.5
Immature	96.5%	95.8%	96.1%	97.2%
Khalal	95.7%	94.9%	95.3%	96.5%
Rutab	94.8%	93.6%	94.2%	95.8%
Tamar	96.1%	94.2%	95.1%	97.0%

(Refer to Figure 10 for the training loss and mean average bar graph).

To deploy, the trained YOLOv11m model was first converted into a slim Python app. The frontend, written in Streamlit, provides a minimalistic interface that allows users to upload images or view real-time detection via a webcam. A

backend runs the inference of the trained YOLOv11m, thereby ensuring a fast, fluid user experience which aligns well with the demands of mobile and embedded deployments (See the Validation and Testing UI in Figure 11).

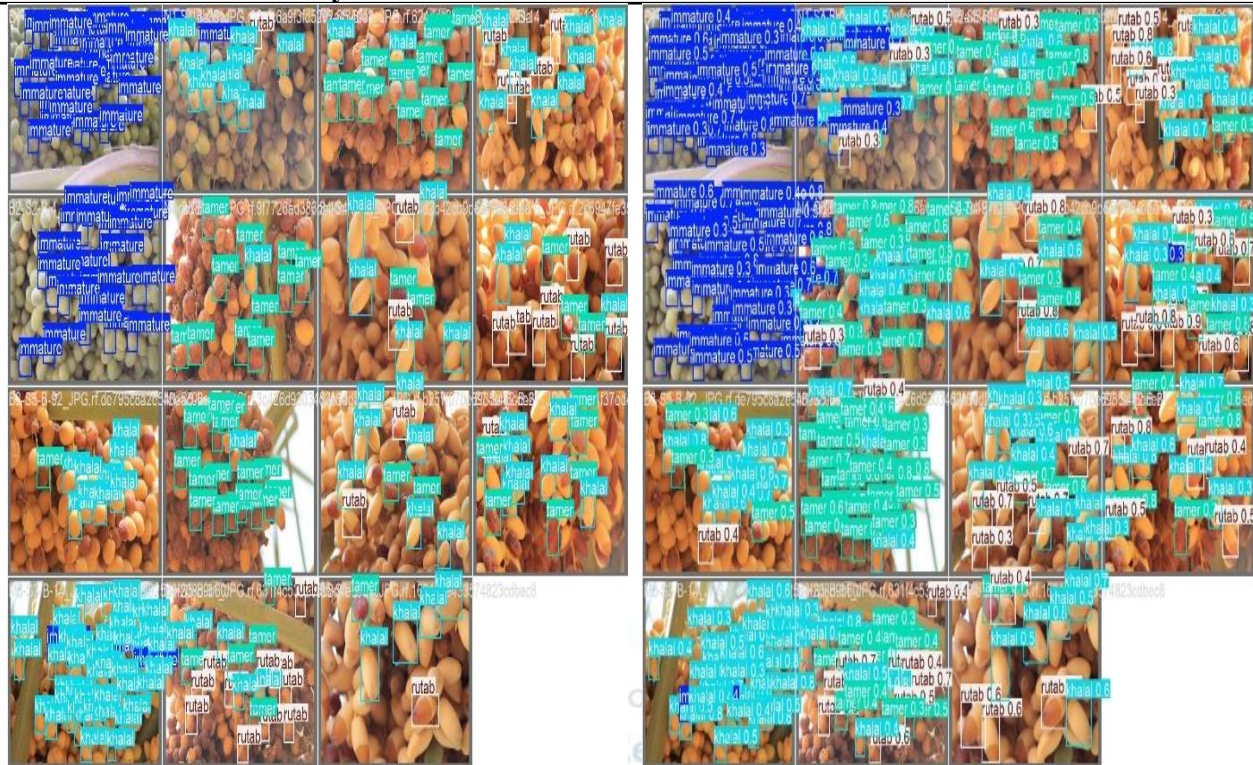


Figure 11. Validation and Testing 1 and Testing 2

6. Why Is Your Work Different?

Our design completely addresses a series of issues that date-farming computer vision systems have

had in the past. Ripen AI's unique benefits over previous methods are listed in Table 12.

Table 12. Advantages of the Proposed System vs. Earlier Studies

Feature	Previous Studies	Proposed System (Ripen AI)
Deep Learning Architecture	CNN, YOLOv3-YOLOv10	YOLOv11m
Object Detection	Limited or Partial	Complete Object Detection
Ripeness Classification	Yes	Yes (4 Stages: Immature, Khalal, Rutab, Tamar)
Real-Time Processing	Rarely Implemented	Fully Supported (55 FPS)
Live Webcam / Upload Inference	Generally, Not Available	Supported
Deployment Interface	Not Provided	Streamlit-Based Interface
Smart Farming / Robotic Harvesting	Mostly Conceptual / Proposed Only	Practical Prototype / Ready for Integration

7. Conclusion and Future Work

The Ripen AI project has successfully demonstrated a deep learning method for date fruits detection and ripeness classification based on ripeness stages using the YOLOv11m

architecture. With the availability of the most recent models and the ability to efficiently manage information via Roboflow, a system that can be relied upon and is easy to use by humans, has been

developed to solve problems in agricultural mechanization.

Despite the results being positive, the model did have certain minor defects; for instance, accuracy could decrease where fruits are significantly masked, the lighting conditions are extremely harsh, or photos are from very untypical angles. As a suggestion, the dataset should be extended to cover a wide range of environmental conditions and be trained with images of greater resolution in the subsequent development. Long-term work would include the development of such a ripeness detection model as part of a totally autonomous robotic harvesting system, thereby enabling robots in the farm to accurately and systematically identify and pick fruits depending on their stage of ripeness.

8. REFERENCES

- Y. Fu and D.-W. Sun, "Advancing fruit quality monitoring with artificial intelligence-augmented non-destructive technologies," *Food Eng. Rev.*, vol. 18, no. 2, Art. no. 9, 2026.
- M. A. Qwaid, M. T. Sarker, S. M. Shawon, and H. T. Zubair, "AI-enabled smart farming framework for sustainable date palm cultivation in arid regions using machine learning and IoT integration," *Sci. Rep.*, 2026.
- M. Gomaa, H. M. Farghaly, A. Darwish, and A. E. Hassanien, "Sustainable agriculture through automated classification of date fruit varieties using deep learning techniques," in *Innovative AI Technologies Driving Sustainable Farming: Strategies for Improving Food Security*, Cham, Switzerland: Springer Nature Switzerland, 2026, pp. 263–284.
- J. Yousaf et al., "Autonomous smart palm tree harvesting with deep learning-enabled date fruit type and maturity stage classification," *Eng. Appl. Artif. Intell.*, vol. 139, Art. no. 109506, 2025.
- A. Maghzaoui, E. Aridhi, S. B. Yahia, S. Bahroun, and A. Mami, "Dataset of Deglet Nour date palm bunches for smart harvesting," *Data Brief*, Art. no. 112217, 2025.
- M. F. Hameed, H. Nazir, M. SamiUllah, and M. Saifullah, "Classification of date fruits for quality control using deep learning," *J. Comput. Biomed. Informatics*, 2025.
- G. Lavanya and S. Pande, "Enhancing real-time object detection with YOLO algorithm," *EAI Endorsed Transactions on Internet of Things*, 2023, doi: 10.4108/eetiot.4541.
- Ultralytics, "YOLOv11 documentation," 2024. [Online]. Available: <https://docs.ultralytics.com/yolov11>
- M. Safran, W. Alrajhi, and S. Alfarhood, "DPXception: A lightweight CNN for image-based date palm species classification," *Frontiers in Plant Science*, vol. 14, p. 1281724, Jan. 2024, doi: 10.3389/fpls.2023.1281724.
- J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.
- J. R. Terven, "A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS," 2024.
- J. Redmon et al., "YOLO9000: Better, faster, stronger," *arXiv preprint arXiv:1612.08242*, 2016.
- Kili Technology, "YOLO algorithm: Real-time object detection from A to Z," 2023.
- B. Dwyer, J. Nelson, T. Hansen, et al., *Roboflow (Version 1.0) [Software]*, 2024. Available: <https://roboflow.com>
- M. Ouhda, Y. Zarouit, and B. Aksasse, "Smart harvesting decision system for date fruit based on fruit detection and maturity analysis using YOLO and K-means segmentation," *Journal of Computer Science*, vol. 9, pp. 1242–1252, 2023, doi: 10.3844/jcssp.2023.1242.1252.

H. Altaheri, M. Alsulaiman, M. Faisal, and G. Muhammed, "Date fruit dataset for automated harvesting and visual yield estimation," *IEEE Dataport*, Feb. 2019, doi: 10.21227/x46j-sk98.

H. Altaheri et al., "Date fruit dataset for intelligent harvesting," *Data in Brief*, vol. 26, p. 104514, Oct. 2019.

H. Altaheri, M. Alsulaiman, and G. Muhammad, "Date fruit classification for robotic harvesting in a natural environment using deep learning," *IEEE Access*, vol. 7, pp. 117115-117133, 2019.

Y. Zarouit, H. Zekkouri, M. Ouhda, and B. Aksasse, "Date fruit detection dataset for automatic harvesting," *Data in Brief*, vol. 52C, p. 109876, 2023, doi: 10.1016/j.dib.2023.109876.

A. K. Maitlo, R. A. Shaikh, and R. H. Arain, "Date fruit dataset for inspection and grading," *Mendeley Data*, V3, 2023, doi: 10.17632/s5zfvsw5kv.3.