FACULTY OF ENGINEERING

REPORT PROPOSAL

EPE 4036 FYP1 PROJECT

Project Title: Mango Fruit Detection from Aerial Image

Offered by: Dr. Mohd Haris Lye Abdullah

Trimester 1, 2023/2024

Name	**Marawan Ashraf Fawzy Eldeib**
Student ID	**1181102334**
Major	**CE**

Table of Contents

Table of Figures:

Chapter 1. Introduction

1.1 Background Overview

The agricultural sector, particularly the cultivation and harvesting of mangoes, being one of the most popular and beloved tropical fruit for its juiciness, delicious taste, and nutritional value. Mango production has heavily relied on human labor and subjective judgment for fruit counting and predicting harvest yields, which is not only time-consuming but also prone to errors. This reliance presents notable limitations in terms of scaling up, effectiveness, and precision. However, recent advancements in agricultural technology, especially the integration of aerial imaging and artificial intelligence, has opened new avenues for innovation.

Modern drones are equipped with high-resolution cameras and deployed to fly in mango orchards, capturing detailed images that are then processed using advanced deep learning frameworks like PyTorch.

PyTorch is a machine learning framework based on the Torch library, used for applications such as computer vision and natural language processing, originally developed by Meta AI and now part of the Linux Foundation umbrella. It is free and open-source software released under the modified BSD license. [1]This open-source machine learning library accelerates the development of complex algorithms such as YOLO (You Only Look Once) and Faster R-CNN, both sophisticated Convolutional Neural Networks (CNNs) designed for object detection.

YOLO, known for its speed, processes images in real-time to detect objects in one look. In contrast, Faster R-CNN, can be implemented using Detectron2 which is Facebook AI Research's next generation library that provides state-of-the-art detection and segmentation algorithms. [2] It is slightly slower but offers potentially higher accuracy through a two-stage detection process that first generates proposals and then classifies them. Both methods utilize convolutional layers to filter and process image data, extracting features crucial for object classification. [3]

In comparing YOLO and Faster R-CNN, this project aims to enhance mango orchard management by automating the detection process, thereby improving accuracy and reducing the need for manual labor. The integration of PyTorch will facilitate the development and training of these deep learning models, driving forward the project's goal to modernize the mango harvesting process.

1.2 Problem Statement

The traditional methods of mango orchard management are time-consuming, labor-intensive, and prone to errors, especially in fruit counting and ripeness assessment. There is a clear need for an automated, precise, and scalable solution to enhance the productivity and accuracy of mango harvesting.

1.3 Project Scope

The proposed project aims to develop innovative solution that leverages drones to capture high-resolution aerial images of mango orchards. Mangoes are oval in shape and can have green, yellow, or red skin. The core of this project is the utilization of advanced deep learning models, specifically YOLO, which are known for their accuracy and efficiency in object detection tasks. By applying these models to the aerial images, the project will automate the process of detecting and counting mango fruits with high precision.

Beyond fruit detection, this project extends its scope to a critical aspect of mango orchard management which is predicting the ripening status of the detected mangoes. This capability to determine fruit maturity in the orchard is crucial for optimizing harvest timing and logistics planning.

Chapter 2. Objectives

The main project objectives for the Mango Fruit Detection from Aerial Image model are:

1. To develop the object detection model for mango fruit detection by using cutting edge deep learning algorithm (YOLO) and compare against Faster R-CNN for accuracy and efficiency.

2. To refine mango detection accuracy and reliability by optimizing model's mean Average Precision (mAP) across various epochs and batch sizes.

3. Predict mango fruit ripeness status across three levels - unripe, ripe, overripe to determine optimal harvest time.

4. Validate models under real-world conditions through extensive field-testing capturing variances like differing light. Also, Test operation with drone moving to emulate production conditions.

5. Share project results and insights through academic publications such as IEEE. To contribute to the advancement of agricultural technology and precision agriculture practices.

Chapter 3. Preliminary Literature Review

3.1 “Deep Fruit Detection in Orchards” Literature Review

S. Bargoti and J. Underwood, "Deep fruit detection in orchards," *2017 IEEE International Conference on Robotics and Automation (ICRA)*, Singapore, 2017, pp. 3626-3633, doi: 10.1109/ICRA.2017.7989417. [4]

3.1.1 Problem Addressed:

The research addresses the need for a precise and dependable system for fruit detection in orchards. This is essential for advanced agricultural tasks, including yield mapping and robotic harvesting. It focuses on the application of the Faster R-CNN object detection framework in accurately identifying fruits such as mangoes, almonds, and apples, and explores the practical aspects of deploying this technology in orchards.

3.1.2 Theories/methods/models used:

The paper utilizes the Faster R-CNN object detection framework. It experiments with different network architectures (ZF and VGG16), explores data augmentation techniques, and evaluates the impact of transfer learning from other orchard datasets.

Data Augmentation: A technique to increase the variability of training data by applying label-preserving transformations (like flipping, scaling, color adjustments), enhancing the model's ability to generalize. Data augmentation significantly reduced the number of training images required.

Transfer Learning: Applying knowledge gained from one task to a different but related task. In this context, it involves using a pre-trained model on one type of fruit or orchard and adapting it to another. Transferring knowledge between orchards had negligible benefits compared to training from scratch with ImageNet features.

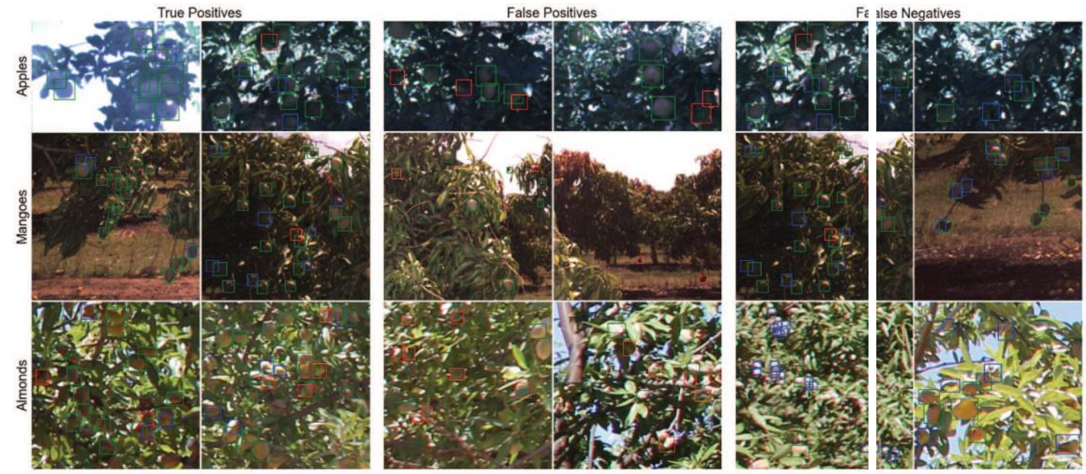

Figure 3.1 visual representation shows outcomes of sample detections for different orchards into three categories.

This figure shows sample detections from the test set for apples, mangoes, and almonds. It visually represents the outcomes classified by the detection model into three categories: true positives, false positives, and false negatives.

True Positives (Green Boxes): These are the instances where the model correctly identifies and locates a piece of fruit in the image. For instance, every green box in the mangoes row indicates a mango that was accurately detected by the model.

False Positives (Red Boxes): These are instances where the model incorrectly identifies an object as a fruit when it is not, or when it detects a fruit that is not actually there. The red boxes in the figure, especially noticeable in the column labeled as 'most false positives', show locations where the model mistakenly identified something as a fruit.

False Negatives (Blue Boxes): These instances occur when the model fails to identify an actual piece of fruit present in the image. In the figure, the blue boxes highlight the fruit that the model missed.

3.1.3 Results:

The study achieved high detection performance (F1-scores over 0.9 for apples and mangoes). It found that data augmentation significantly reduced the training data required, while transfer learning from other orchards showed minimal performance gains.

Also, the image below shows, show detection of all mangoes over a mango tree using Tiled Faster R-CNN. The example image contains 54 true positive detections (in green), 2 false negative detections (in blue) and no false positives.

Figure 3.2 detection of all mangoes over a mango tree

3.1.4 Strengths:

- High detection accuracy with F1-scores over 0.9.
- In-depth analysis of training data requirements.

- Effective application of data augmentation techniques.

- Detailed analysis of Faster R-CNN for orchard data.

3.1.5 Weaknesses:

- The study utilized a ground-based robotic vehicle for image capturing, which may have limitations in accessing certain areas and angles in an orchard compared to aerial drones. Drones can potentially achieve higher accuracy due to their ability to get closer to the fruit canopy.

- Limited exploration of transfer learning benefits.

- It primarily examines the Faster R-CNN framework, potentially overlooking other emerging technologies in object detection.

3.1.6 Applications to Current Project:

- The dataset from the University of Sydney's Australian Centre for Field Robotics will be pivotal for training and testing our deep learning models, offering real-world conditions. [5]

- Employing data augmentation, as in the study, will improve our model's performance under different scenarios.

- Transfer learning insights will guide us in adapting the model specifically for mango orchards.

3.2 "Visual detection of green mangoes by an unmanned aerial vehicle in orchards based on a deep learning method" Literature Review.

Juntao Xiong, Zhen Liu, Shumian Chen, Bolin Liu, Zhenhui Zheng, Zhuo Zhong, Zhengang Yang, Hongxing Peng, Visual detection of green mangoes by an unmanned aerial vehicle in orchards based on a deep learning method, Biosystems Engineering, Volume 194, 2020, Pages 261-272, ISSN 1537-5110,https://doi.org/10.1016/j.biosystemseng.2020.04.006.
(https://www.sciencedirect.com/science/article/pii/S1537511020300970) [6]

3.2.1 Problem Addressed:

This research focuses on developing a method for visually detecting green mangoes in orchards using UAVs. It addresses the challenge of estimating mango fruit numbers accurately, which is vital for efficient orchard management.

3.2.2 Theories/methods/models used:

The study employs the YOLOv2 (You Only Look Once version 2) model for quick mango detection. It involves capturing mango images via UAVs, manually marking these images to create training and test datasets and fine-tuning the YOLOv2 model parameters based on these datasets.

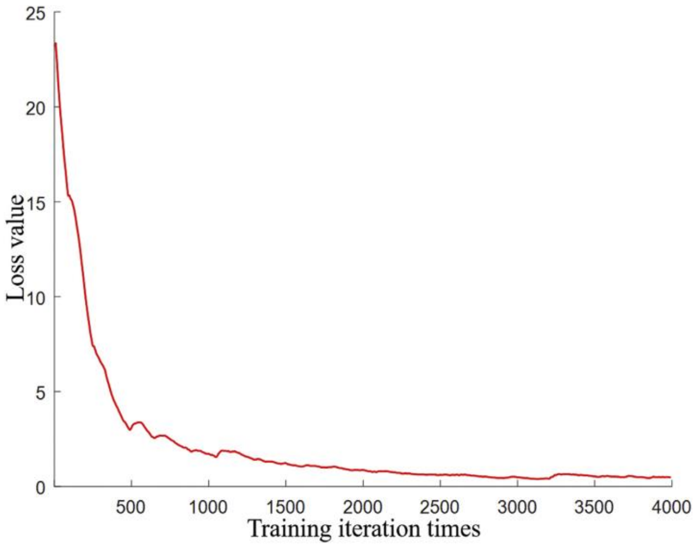

Figure 3.3 The change of network loss value

The graph shows that as the model trains, the loss value decreases, which indicates the model's improving performance in detecting mangoes. Initially, the loss drops sharply, and after several iterations, it stabilizes, suggesting the model has learned effectively from the training data.

Also, the confusion matrix is used to calculate the precision and recall, essential metrics for evaluating the model's performance. Precision is the ratio of true positive detections to the sum of true positives and false positives, while recall is the ratio of true positives to the sum of true positives and false negatives. These metrics provide insight into the accuracy and thoroughness of the model's detection capabilities.

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

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

Figure 3.4 confusion matrix formulas used to calculate precision and recall.

Where TP is true positive, FN is false negative, and FP is false positive.

3.2.3 Results:

The study demonstrated effective mango detection, achieving a precision rate of 96.1% and a recall rate of 89.0%. An experiment for estimating the actual number of green mango fruits was also conducted, showing a low estimation error rate, indicating the model's effectiveness in practical applications.

Figure 3.5 The detection results of mango images

3.2.4 Strengths:

- High precision and recall rates in mango detection.
- Effective use of UAV imagery for data collection.

- Detailed exploration of YOLOv2 for orchard fruit detection.

3.2.5 Weaknesses:

- Dependence on YOLOv2, which is an earlier iteration of YOLO technology which does not include the latest improvements in object detection techniques.

3.2.6 Applications to Current Project:

- The methodologies and findings of this study can guide the development of UAV-based fruit detection systems in your project.

- Insights from the model's performance can inform the selection and tuning of deep learning models for efficient mango detection in orchards.

Chapter 4. Methodology

This chapter presents the methodological approach that will underlie the mango fruit detection from aerial images, overing each stage from the initial data gathering to concluding real-world validation of the developed model.

4.1 Data Collection and Annotation:

The first step is to find a dataset that accurately represents the subject matter of the project, in this case, images of mangoes on trees. The dataset from the University of Sydney's Australian Centre for Field Robotics provides a robust foundation for training and validating the deep learning models. [5] To enhance the dataset further, additional imagery will be gathered using drones, capturing videos of mango orchards. These videos will be converted into still frames to augment the dataset.

All images will undergo annotation via Roboflow, a comprehensive tool for data annotation and pre-processing. Roboflow will facilitate the labeling process, adding context to the raw image data to enable the machine learning model to learn from it effectively. [7] Once annotated, the dataset will be divided into training (70%), validation (20%), and testing (10%) sets, a common distribution that allows for comprehensive training and unbiased evaluation of the model's performance.

4.2 Model Selection and Configuration:

The project will employ YOLOv8, leveraging the PyTorch framework for its implementation. YOLOv8 is the latest in the YOLO series, chosen for its exceptional balance of speed and accuracy. The specific variant, likely YOLOv8m, will be selected based on the trade-off between computational efficiency and detection performance as visualized in the provided graphs, which compare different YOLO versions in terms of size, speed, and COCO mAP (val) scores.

▼ Detection (COCO)

See Detection Docs for usage examples with these models trained on COCO, which include 80 pre-trained cla

Model	size (pixels)	mAP^{val} 50-95	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	params (M)	FLOPs (B)
YOLOv8n	640	37.3	80.4	0.99	3.2	8.7
YOLOv8s	640	44.9	128.4	1.20	11.2	28.6
YOLOv8m	640	50.2	234.7	1.83	25.9	78.9
YOLOv8l	640	52.9	375.2	2.39	43.7	165.2
YOLOv8x	640	53.9	479.1	3.53	68.2	257.8

Figure 4.1 Comparison of different YOLOv8 models [8]

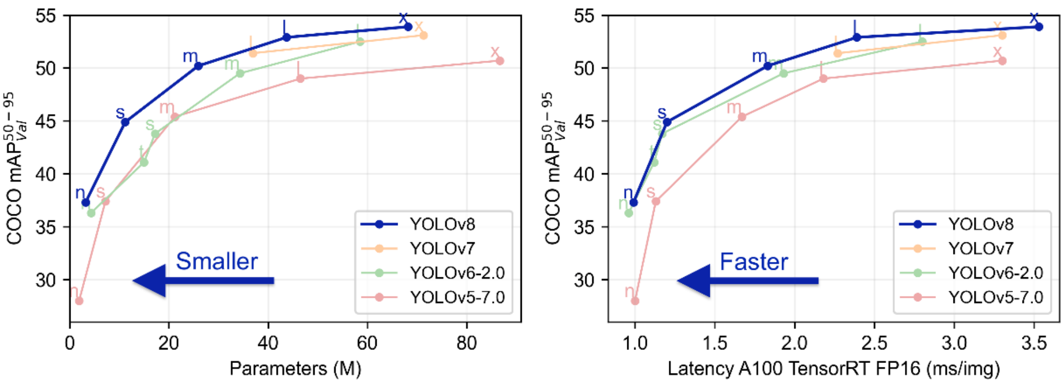

Figure 4.2 Comparison of different YOLO models [8]

The annotated dataset will find its way to Google Colab, a cloud-based platform that facilitates the model training with its accessible computational ability. [9]

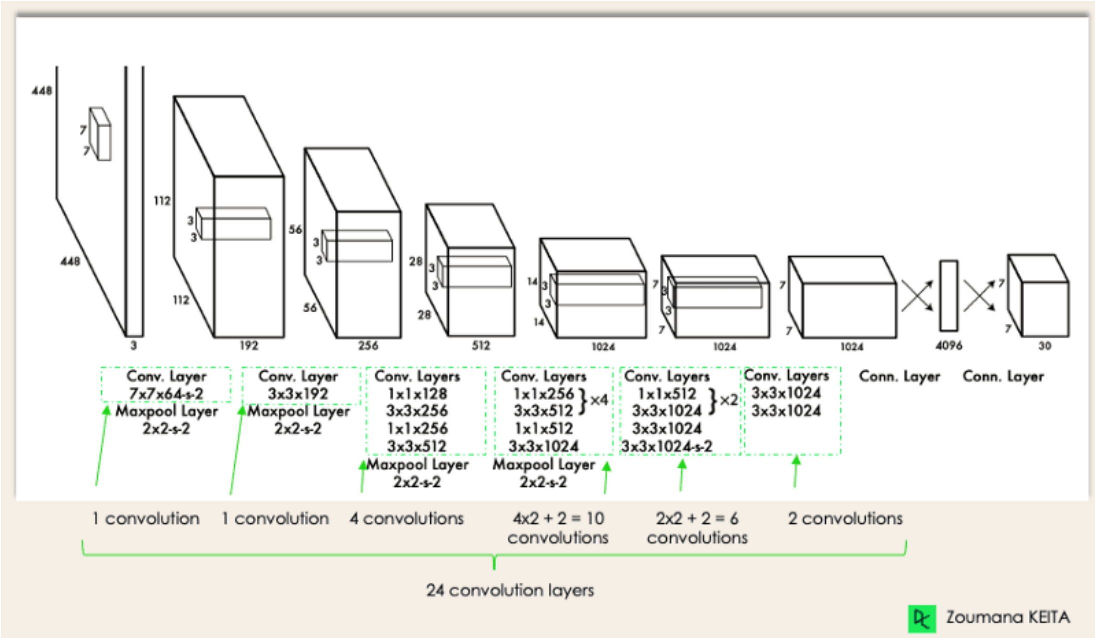

Figure 4.3 YOLO Architecture diagram [10]

4.3 Training and Model Evaluation:

Our training approach will conduct multiple passes of the dataset, known as epochs, through the neural network, refining the model with each cycle to ensure accuracy and minimize errors. Careful monitoring will prevent overtraining and maintain model reliability. For assessment, we'll use robust Python libraries to evaluate our model's precision and recall, aiming for a high F1-score, a mark of quality for accurate and comprehensive detection. Refinements will be ongoing to perfect the model based on these evaluations.

The numpy library will handle complex numerical operations, especially those involving multi-dimensional arrays, which are crucial for image data processing. [11] Pandas will manage our dataset, allowing for efficient data manipulation and preparation. [12] Finally, matplotlib will come into play for visualizing the model's performance metrics, such as loss and accuracy, in a clear and interpretable manner. [13] Together, these tools will help us measure, evaluate, and improve our model to ensure high precision in detecting and classifying mango ripeness.

4.4 Ripeness Prediction Model Development:

Beyond just mango detection, determining the ripeness of mangoes is pivotal for harvest planning. To achieve this, similar to the object detection model, the ripeness prediction model will be trained using a set of labeled images that include various ripeness stages. The ripeness prediction model will be integrated with the primary detection model. The combined system will detect mangoes and simultaneously predict their ripeness- unripe, ripe and overripe providing a comprehensive analysis of the orchard's condition in real-time.

4.5 Project Milestone

Figure 4.4 Project Gantt Chart

4.6 Project Budget

The project will utilize resources provided at no cost. This includes:

Drone for Aerial Imaging: Supplied by the faculty for capturing images of the orchards.

Model Development Platform: Google Colab's free version will be used for developing, training, and testing the machine learning model.

Additional Costs: Currently, there are no additional costs anticipated for the completion of this project.

This section acknowledges the resources provided and clarifies that no extra budget is required for this phase of the project.

References

[1] "pytorch," [Online]. Available: https://pytorch.org/.

[2] "Detectron2 by Facebook AI," [Online]. Available: https://ai.meta.com/tools/detectron2/.

[3] "faster-r-cnn," [Online]. Available: https://blog.paperspace.com/faster-r-cnn-explained-object-detection/.

[4] S. &. U. Bargoti, ""Deep fruit detection in orchards,"," *IEEE International Conference on Robotics and Automation (ICRA),* pp. pp. 3626-3633, 2017.

[5] "The University of Sydney's ACFR Multi-fruit Dataset," [Online]. Available: https://data.acfr.usyd.edu.au/ag/treecrops/2016-multifruit/.

[6] Zhen Liu, Shumian Chen, Bolin Liu, Zhenhui Zheng, Zhuo Zhong, Zhengang Yang, Hongxing Peng, "Visual detection of green mangoes by an unmanned aerial vehicle in orchards based on a deep learning method," *Biosystems Engineering,* vol. 194, no. 1537-5110, pp. 261-272, 2020.

[7] "roboflow," [Online]. Available: https://roboflow.com/.

[8] "YOLOv8 Documentation," [Online]. Available: https://docs.ultralytics.com/.

[9] "Google Colab," [Online]. Available: https://colab.research.google.com/.

[10] "yolo architecture," [Online]. Available: https://www.datacamp.com/blog/yolo-object-detection-explained.

[11] "numpy," [Online]. Available: https://numpy.org/.

[12] "pandas," [Online]. Available: https://pandas.pydata.org/.

[13] "matplotlib," [Online]. Available: https://matplotlib.org/.