Optimizing Apple Orchard Management with Machine Learning: Diagnostic Models for Harvesting

by

Alisher Zhakenov

Submitted to the Department of Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Computer Science

at the

NAZARBAYEV UNIVERSITY

June 2025

Author . . . .
Department of Computer Science
April 11

Certified by . . . .
Askar Boranbayev
Professor
Thesis Supervisor

Accepted by . . . .
Yelyzaveta Arkhangelsky
Dean, School of Engineering and Digital Sciences

Optimizing Apple Orchard Management with Machine Learning: Diagnostic Models for Harvesting

by

Alisher Zhakenov

Submitted to the Department of Computer Science
on April 11, in partial fulfillment of the
requirements for the degree of
Master of Science in Computer Science

Abstract

Efficient management of apple orchards is crucial for ensuring optimal fruit yield and quality. However, managing apple orchards has various factors such as pest control, extreme weather effects, and predicting the best harvest time, which remains a challenge for orchard managers. This thesis aims to apply machine learning techniques to optimize orchard management, focusing on diagnostic models that analyze the intrinsic state of individual trees. By using images of the trees, the model can separately identify the condition of each main part of the apple tree (apples and leaves) allowing it to generate a comprehensive description of the tree's overall health. This model can potentially reduce costs, increase productivity, and promote sustainable orchard practices. The research demonstrates how machine learning can be used in agriculture to assess the individual state of trees in orchards, allowing managers more precise control over orchard maintenance.

Thesis Supervisor: Askar Boranbayev
Title: Professor

Acknowledgments

I would like to thank Nazarbayev University for offering valuable machine-learning courses that have greatly shaped my understanding of this field. The knowledge I gained through these courses contributed significantly to the development of this thesis.

Also, special appreciation to my supervisor, Professor Askar Boranbayev, whose guidance, patience, and advice have been a great help throughout this thesis.

I am also incredibly grateful to my family for their constant encouragement and belief in me. Their support made this thesis possible.

Contents

1 Introduction **13**

2 Related works **17**

2.1 Object detection 17

2.2 Apple quality assessment 19

2.3 Leaf disease identification 21

3 Methodology **25**

3.1 YOLOv5 26

3.1.1 Dataset Preparation and Preprocessing 26

3.1.2 Model Architecture and Training 27

3.2 EfficientNet Model 28

3.2.1 Dataset Preparation and Preprocessing 28

3.2.2 Model Architecture and Training 29

3.3 ResNet 29

3.3.1 Dataset Preparation and Preprocessing 29

3.3.2 Model Architecture and Training 30

3.4 Integrated Models 31

3.4.1 Dataset Preparation and Preprocessing 31

3.4.2 Model Architecture 32

4 Results **35**

4.1 YOLOv5 performance on Object Detection 35

4.2 EfficientNet Performance on Apple Classification 36
4.3 ResNet Performance on Leaf Classification 37
4.4 Integrated Model Performance 39

5 Conclusion **41**

List of Figures

3-1 Architecture 25

3-2 Merged dataset samples 27

3-3 Apple dataset samples 28

3-4 Apple dataset samples 30

3-5 Tree dataset samples 32

4-1 EfficientNet accuracy and loss on training 36

4-2 Resnet accuracy and loss on training 38

4-3 Examples of bad object capture 39

List of Tables

2.1 Apple Counting and Detection Models 18
2.2 Fruit Quality Assessment Models 19
2.3 Leaf Disease Identification Models 22

4.1 Evaluation results for the YOLOv5 model on the test set. 35
4.2 EfficientNetB0 evaluation metrics on the apple classification test set. 37
4.3 ResNet evaluation metrics on the leaf classification test set. 38
4.4 Accuracy information for healthy and diseased apple images. 39

Chapter 1

Introduction

Apple orchards constitute a significant sector of global agriculture, contributing to the food chain along with the economy of many regions. As one of the most widely cultivated and utilized fruits, apples are not just prized for their taste but also for their nutritional property [1]. In the last several years, the heightened interest in healthy lifestyles and sustainable food systems has also increased the demand for apples, placing pressure on orchard managers to increase harvest while maintaining fruit quality. However, this task is not an easily achievable feat since it requires careful observation of various factors like climate change, soil fertility, water supply, pest infestation, and tree health. All of these variables can significantly affect the yield and quality of the crop requiring precise and immediate decision-making in orchard management.

Traditionally, the operation of orchards has been dependent on human experience and visual, on-site assessment. Orchards have usually been inspected up and down the rows by farm managers and growers, briefly inspecting leaves for pathogens, and pests, and assessing overall tree health. While these strategies have been partly successful, they are inherently labor-intensive and contain significant flaws. Manual inspection is particularly difficult in big orchards, where hundreds or thousands of trees need to be inspected. Moreover, human vision is subjective and error-prone, especially for the detection of subtle symptoms of disease which may later affect many trees and result in low yields, higher costs, and irreparable harm to the ecological balance of the orchard.

In recent years, advances in technology have started to change agricultural practice, with it bringing highly advanced tools to address these issues. Technologies like drones mounted with multispectral cameras, satellite imagery, and Internet of Things (IoT) sensors are powerful instruments for tracking large-scale agricultural operations [2]. These technologies empower farmers to gather information about environmental conditions, crop development trends, and potential stressors, thereby assisting in immediate decision-making. For instance, drones can capture images of whole orchards from the air, offering information about infections and revealing their area. In the same way, IoT sensors inserted into the soil can provide real-time measurements of moisture levels, nutrient levels, and other relevant parameters. But despite their groundbreaking potential, these technologies are primarily concerned with macro-level monitoring, providing bulk data on large sections of the orchard but not on individual trees. This limitation implies that the early signs of pest infestations, disease, or nutrient deficiencies at the tree level might go unnoticed until the problem has already spread, thus diluting the impact of early interventions.

To address this crucial gap, this thesis proposes the development and use of a machine-learning model that will keep track of and analyze the health of every tree in an orchard. The proposed model is built around the two most crucial aspects of apple trees: apples and leaves, which are significant predictors of tree health and production. Leaves, in particular, displays visible signs of stress, disease, or nutrient deficiencies well before the effects are apparent in the fruit. Apples, on the other hand, are the end product, and their quality overall are the product of the accumulative effect of environmental and physiological factors on the tree. Applying advanced image processing techniques and machine learning, the model will inspect apples and leaves one by one, identifying blights, spots, and rusts. The resulting data will provide detailed information regarding the well-being of each tree, enabling the orchard managers to take specific action and prevent minor problems from escalating into major problems. There are several advantages to the use of this method. First, it offers precision which allows for localized interventions by the managers of the orchards, such as spraying pesticides to specific areas or adjusting irrigation time schedules for individual trees, thereby optimizing the utilization of resources and minimizing the impact on the environment. Second, early detection of problems via the model facilitates immediate management that prevents problems from escalating into serious problems that are able to affect the entire orchard. Third, automation conserves labor and reduces on human mistakes because data processing is mechanized, thus everything is processed through it automatically. Overall benefits have the potential to maximize productivity in an orchard, enhance fruit quality, and promote sustainable cultivation practices.

Major objectives:

- To develop YOLOv5 for object detection
- To develoop EfficientNet for apple quality assessment
- To develop ResNet model for disease identification from leaf images

The rest of the document is organized as; Section 2 is a literature review where existing studies are compared and assessed. In section 3, the methodology is presented, and functions of all 3 models are described. In section 4 results of the tests are shown. And lastly, section 5 is a conclusion.

Chapter 2

Related works

Apple orchard management involves several tasks that are integral to agricultural productivity, such as disease diagnosis, assessment of fruit quality, and monitoring overall tree health. Various conventional methods are very labor-intensive; however, with recent progress in image processing, machine learning (ML), and deep learning (DL), all these processes have been automated by innovative techniques. This literature review assesses and compares key studies that address challenges of object detection, disease detection, and fruit quality classification.

2.1 Object detection

Object detection plays a crucial role in modern orchard management, particularly in tasks such as automated fruit picking, yield estimation, and health monitoring. Various deep learning-based methods have been employed for apple detection in a natural environment. Among them, YOLOv5 has proven to be the most efficient and precise solution due to its optimized architecture, improved speed, and higher detection accuracy.

Table 2.1: Apple Counting and Detection Models

Ref	Model Used	Performance
[3]	YOLOv5 (Improved YOLOv5 with BottleneckCSP + SE)	91.48% accuracy, 86.75% mAP, F1 87.49%
[4]	YOLOv5 (Depth separable convolution + attention mechanisms)	96.79% mAP, 15.37% improvement in detection speed
[5]	YOLOv3, Faster R-CNN	89% accuracy

Yan et al. [3] came up with a enhanced YOLOv5 model to address the challenge of apple detection under leaf coverage. With improvement in BottleneckCSP and incorporation of an SE module, the model achieved a 86.75% mAP, which surpasses YOLOv3, YOLOv4, and EfficientDet-D0 models. Apart from enhancing detection accuracy, computational complexity reduced significantly, making it very appropriate for application in real time for orchard management. In parallel, Wang et al. [6] upgraded YOLOv5 with the addition of MobileNetv2's inverted residual convolution block, improving both efficiency and model speed. The approach reduced the size of the model by 57% and sped up detection by 27.6%, enabling high-speed detection of apples at a speed of 90 frames per second. The study confirmed YOLOv5 to be faster than YOLOv8s and hence deserves to be utilized for high-performance detection of apples. Other improvements in YOLOv5 were made by Zhijun et al. [4], who incorporated depth separable convolution and attention mechanism to enhance detection. Their model improved the detection speed by 15.37% and mAP by 96.79%, which illustrates the efficiency of YOLOv5 in yield estimation of orchard crops. Their observations showed that YOLOv5 has high accuracy even at high speeds and under varying light conditions, reinforcing its strength in real farm conditions. Other object detection algorithms, such as Faster R-CNN and YOLOv3, have also been applied to apple detection [5]. Although these models perform well under occlusion and illumination changes, they lack computational efficiency and processing speed. Despite newer versions of YOLOv3 being available, they still fall behind YOLOv5 in terms of accuracy and real-time detection. YOLOv5 stands out as the most optimal model for apple object detection in orchard management based on the reviewed research. Its highly optimized architecture allows for efficient and effective apple detection, which is essential for robotic picking and orchard monitoring. Compared to earlier versions of YOLO and other deep learning models, YOLOv5 offers the best combination of speed, accuracy, and computational efficiency. Given these advantages, YOLOv5 will be used as the underlying object detection model in this research to track apple orchards and provide real-time, precise analysis of fruit conditions.

2.2 Apple quality assessment

In recent years, convolutional neural networks (CNNs) have become popular for automating various tasks in precision agriculture, particularly for the classification of vegetables and fruits and evaluation of their quality. Of these models, EfficientNet and its variants have emerged as top performers due to their ability to sustain high accuracy at the cost of computational overhead, enabling them to perform well on both cloud-based and edge devices in agricultural settings.

Table 2.2: Fruit Quality Assessment Models

Ref	**Model Used**	**Performance**
[7]	EfficientNetV2 (Dual Model with AugMix, CutMix, MixUp)	99.49% classification, 99.42% grading
[8]	EfficientNet-Lite4	88.89% accuracy
[9]	CA-EfficientNet-CBAM	95.12% accuracy
[10]	EfficientNet	96.8% accuracy, 96.4% F1
[11]	EfficientNet (Raspberry Pi based real-time grading)	99.2% apples, 98.6% bananas

Aldakhil and Almutairi [7] applied EfficientNetV2 in a double-model architecture for multi-fruit classification and grading of quality on a dataset from FruitNet. By applying same-domain transfer learning, their strategy attained test accuracies of 99.49% for classification and 99.42% for grading. They also added data augmentation techniques such as AugMix, CutMix, and MixUp to remove class imbalance and demonstrate the model's robustness and flexibility under different agricultural environments. Dionisio et al. [8] demonstrated the viability of using EfficientNet-Lite 4 on low-power embedded devices (Raspberry Pi 4) for classifying Philippine strawberries into pre-specified classes. The model achieved 100% accuracy for "Extra Class" strawberries while showing an overall classification accuracy of 88.89%, highlighting EfficientNet's ability to function efficiently even with limited computational resources. Wen and He [9] introduced a customized model, CA-EfficientNet-CBAM, to classify six vegetables. The model integrated channel and spatial attention mechanisms to enhance feature focus and reduce training time. In comparison with VGGNet16, ResNet50, and DenseNet169, the model produced the highest accuracy of 95.12% with the fewest parameters, justifying EfficientNet's advantage in efficacy and accuracy. Furthermore, Nigam [10] provided a comparison of performance of numerous deep learning models and inferred that EfficientNet performed better than ResNet and baseline CNNs with 96.8% accuracy, 96.3% precision, 96.5% recall, and 96.4% F1-score. Although EfficientNet was trained for longer duration, it was also proved to possess the best ROC-AUC and specificity and hence most suitable for grading assignments and indicating the need for future optimizations in order to increase scalability. Ismail and Malik [11] developed a real-time fruit grader by combining EfficientNet with a Raspberry Pi-based machine vision system. EfficientNet achieved 99.2% for apples and 98.6% for bananas, outperforming ResNet, DenseNet, MobileNetV2, and NASNet. Physical sample-based real-time testing yielded 96.7% for apples and 93.8% for bananas, proving the model's efficacy in real orchard settings. They also extended the system further by employing stacking ensemble techniques, which yielded marginal improvements and proved EfficientNet-based models to be scalable. Collectively, these contributions provide definitive evidence for the use of EfficientNet as the preferred architecture for apple quality assessment. Its ability to sustain with high performance and lightweight architecture, flexibility through transfer learning, and ability to perform better than baseline models on a variety of agriculture datasets make it an ideal choice for what will be required of an orchard health monitoring system. EfficientNet's proven high-performance ability in both high-resource and edge-computing settings makes it an incredibly capable and scalable model for apple quality diagnostics in orchards.

2.3 Leaf disease identification

With a rising demand for precision agriculture and proper and computerized diagnosis, a variety of models of deep learning have been established for plant disease diagnosis, most specifically for the leaves since these are visible enough to identify unique symptoms. The diseases of apples and trees are heavily affecting both quality and quantity, and right and on-time diagnosis is a crucial process for orchard management. Among the previous research studies, Chao et al. [12] proposed a DenseNet-Xception hybrid DCNN model for the diagnosis of ATLD. By making use of the dense feature reuse capability of DenseNet and depthwise separable convolution of Xception, the model achieved a decent accuracy rate of 98.82%, highlighting deep learning's high-level feature extraction capability from dense agricultural data sets. To compare, the INAR-SSD model by Jiang et al. [13] was introduced in which inception modules from GoogLeNet along with rainbow concatenation are implemented for identifying five common diseases real-time. Although its precision was reduced (78.80% mAP), the model had 23.13 FPS working and was consequently suitable for massive, real-time orchard monitoring. These works highlight the variety of neural architectures employed for plant disease diagnosis—each involving a compromise among accuracy, complexity, and inference speed.

Table 2.3: Leaf Disease Identification Models

Ref	**Model Used**	**Performance**
[12]	DenseNet-Xception	98.82% accuracy
[13]	INAR-SSD (GoogLeNet based)	78.80% mAP; 23.13 FPS
[14]	ResNet-18	98.5% accuracy
[15]	ResNet-34	99.4% test accuracy
[16]	ResNet-18	98.5% accuracy
[17]	ResNet-50 + CA + WAMFF	98.32% accuracy

Among them, residual networks (ResNets) have been a very viable option due to their capacity to train deeper architectures without loss of accuracy, owing to their skip connections and residual learning model. Li and Rai [14] explored the classification of apple leaf diseases namely grey-spot, black star, and cedar rust by comparing traditional SVM-based segmentation methods with CNNs such as VGG and ResNet. Their experiments proved that ResNet-18, being a lighter member of ResNet family, was performing better with 98.5% accuracy and performed a good compromise between depth and computationally efficiency. The similar trend continued in Kumar et al. [15], as they employed deeper ResNet-34 architecture and trained on gigantic open dataset containing 15,200 leaf images of a variety of crops. At 99.40% test accuracy, the study confirmed the effectiveness of ResNet models in generalized disease detection of crops on different plant species and disease types. Similarly, Sirenjeevi et al. [16] confirmed these findings with ResNet-18, achieving 98.5% accuracy in apple leaf disease classification, which shows its reliability and consistency in low-resource but high-efficient applications. On this basis, Zhang et al. [17] proposed another notable enhancement by incorporating coordinate attention (CA) modules and weight-adaptive multi-scale feature fusion (WAMFF) to enhance ResNet-50. With this new version, a precision of 98.32% was achieved on the AppleLeaf9 dataset—4.58% better than the barebones ResNet-50 and even better than many other popular, widely used mainstream networks such as VGG16 and DenseNet. These improvements allowed for better lesion localization and generalization, making

ResNet not only a viable candidate but also an extremely versatile and up-to-date solution for identifying diseases and insects in apples.

Chapter 3

Methodology

This thesis proposal focuses on an approach to optimize the management of apple orchards by analyzing images of apples for their ripeness and leaves for diseases using integrated machine learning models and statistical techniques. In essence, the entire process can be done with three major components: YOLOv5 for detecting objects, EficientNet for analyzing the ripeness and quality of apples, and ResNet for the detection of leaf diseases. The outputs from YOLOv5 and EficientNet components are sent to the custom database to map into database information and produce comprehensive results of tree diagnosis. The architecture of the model is shown in Fig 3-1. Below, each component is explained in detail, along with the integration process.

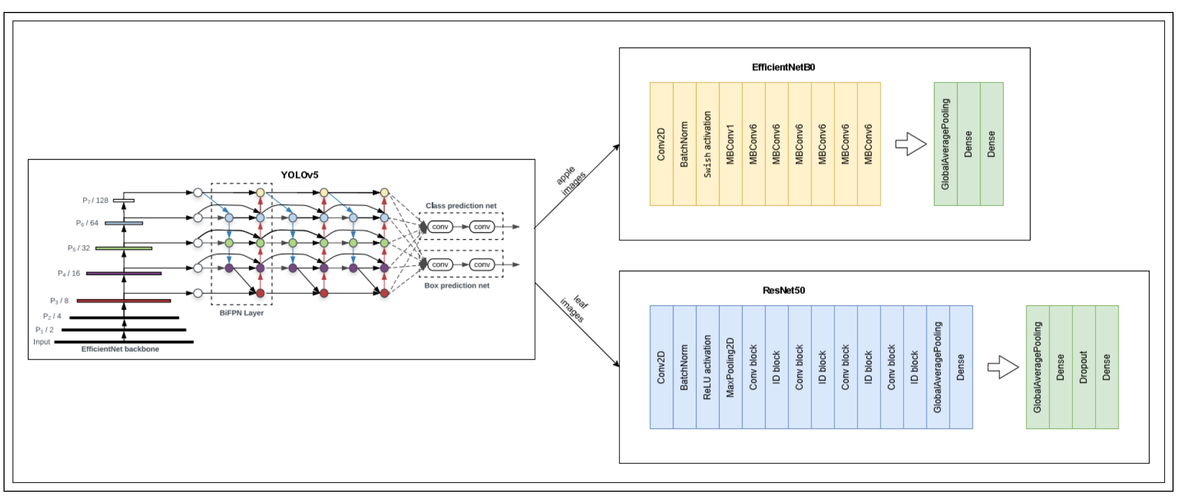

Figure 3-1: Architecture

3.1 YOLOv5

The experiment employs YOLOv5 for leaf and apple detection, and the model is fine-tuned on a healthy and diseased leaf dataset. COCO dataset pre-trained YOLOv5 was utilized to train the model on a new object class, 'leaf,' and 'apple' by modifying the annotation labels of the datasets.

3.1.1 Dataset Preparation and Preprocessing

There were used 2 separate datasets, one for leaves while other for apples. There are 13,540 images of leaves in the first dataset, which are categorized into Healthy and Sick [18]. For object detection, all images were manually relabeled to one class, 'leaf,' so the YOLO model can detect and classify leaves regardless of whether they are healthy or unhealthy. The data were separated into three subsets: the training set has 13,110 images (97%) for fine-tuning, the validation set has 21 images (0%) for monitoring training performance, and the test set has 409 images (3%) for final testing. Annotations were rescaled to YOLO format to specify bounding box coordinates and class label ('leaf'). Rotation, change in brightness, and adding noise were performed as transformations to enrich the dataset so that it favors generalization. The second dataset focuses on apples and contains 1,814 images, categorized as apples [19]. The dataset is divided into three subsets: the training set with 1,314 images (72%) for training, the validation set with 248 images (14%) for fine-tuning and monitoring performance during training, and the test set with 252 images (14%) for final evaluation. Like in the leaf dataset, the labels for apples were also converted into the YOLO format with bounding box coordinates and the class label ('apple'). Rotation, brightness adjustment, and noise addition data augmentation techniques were employed to enhance the model's generalization ability and prevent overfitting. This dataset is used to enable the YOLO model to detect and classify apples and distinguish apples from other objects within the image.

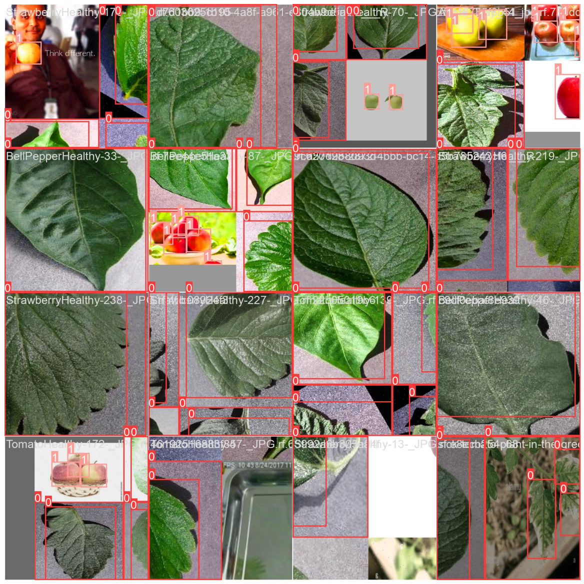

Figure 3-2: Merged dataset samples

3.1.2 Model Architecture and Training

YOLOv5 model, pre-trained on weights was fine-tuned for the detection of two classes: 'apple' and 'leaf'. The model architecture, which was originally trained on the COCO dataset, was modified by adding a new detection head to accommodate the new 'leaf' and 'apple' classes while retaining the learned features of the base classes. The training process was directed to alter only the detection layers while the early convolutional layers were kept frozen in order to retain low-level features learned from the original dataset. To learn efficiently, the learning rate was kept at 1e-4 and the batch size was kept at 16 while it was trained for 50 epochs. Mosaic augmentation was also enabled for the detection of smaller objects more efficiently, which made the model stronger.

The dataset was constructed by merging image and label files from different datasets for the 'apple' and 'leaf' classes, with label adjustments to assign the 'apple' class to index 1 and the new 'leaf' class to index 0. The model was trained on the Darknet framework and the stochastic gradient descent (SGD) optimizer with a confidence threshold of 0.25. The loss function included localization loss, objectness loss, and classification loss to ensure the model's accuracy in object detection. The training was conducted with custom data.yaml configuration file, in which paths to the merged dataset for training, validation, and testing were indicated, and the two classes, 'leaf' and 'apple', were specified.

3.2 EfficientNet Model

The apple images were categorized as fresh and rotten based on the EfficientNetB0 model, a light but effective deep learning model. The steps involved dataset preparation, preprocessing, model training, and evaluation to obtain correct classification.

3.2.1 Dataset Preparation and Preprocessing

The data set consisted of 1,187 images, which were divided systematically into training (70%), validation (20%), and test (10%) subsets [20]. Each image was marked as fresh or rotten, and these were utilized as ground truth for model training. Path to images and corresponding labels were read from CSV files, and labels were one-hot encoded for categorical classification.

To ensure consistency, all the images were resized to 224×224 pixels to follow EfficientNetB0 input specifications. Preprocessing specific to EfficientNet was performed to scale pixel values to improve model performance. The data was then converted to TensorFlow datasets for efficient batching, shuffling, and augmentation-free training.

Figure 3-3: Apple dataset samples

3.2.2 Model Architecture and Training

The EfficientNetB0 model was utilized as a feature extractor and contrary to typical transfer learning setups, this implementation did not use pre-trained ImageNet weights, allowing the model to learn features specific to the apple classification task from scratch. The base model was loaded without its top classification layers, and a custom classification head was appended. A global average pooling layer and a fully connected layer of 256 neurons with L2 regularization to prevent overfitting were used. The final classification layer used a softmax activation function to predict the probability of the two classes.

Throughout the training, the EfficientNetB0 base model was frozen for initial epochs to retain pre-trained features while the remaining layers were trained. The Adam optimizer with learning rate 1e-4 was used, and categorical cross-entropy as the loss function. The model was trained for 100 epochs, using class weighting to counteract potential imbalances in the dataset.

Once trained, the model was tested on the test set to examine its ability to generalize. Test dataset, having unseen images, was utilized for the calculation of accuracy and loss measurements. Performance metrics at the final stage supplied an objective score on how efficient the model can be in terms of classifying new and decayed apples for reliable use on the real field.

3.3 ResNet

Leaf image classification was carried out using a modified ResNet50 deep model. The process involved dataset preparation, preprocessing, model training, and testing to obtain precise disease diagnosis.

3.3.1 Dataset Preparation and Preprocessing

The data set comprised 1,230 images categorized into five disease classes [21]. It was partitioned into training (79%), validation (12%), and test (9%) subsets for model training and testing purposes. One-hot encoding format was utilized for labeling each image to support multi-class classification.

Every one of the images was resized to 224×224 pixels in order to align with ResNet50's input size. Pixel normalization was done as part of the preprocessing activities via the ResNet preprocessing function in order to achieve maximum feature extraction capacity. The dataset was converted into TensorFlow datasets using optimal loading, batching, and prefetching for efficient training.

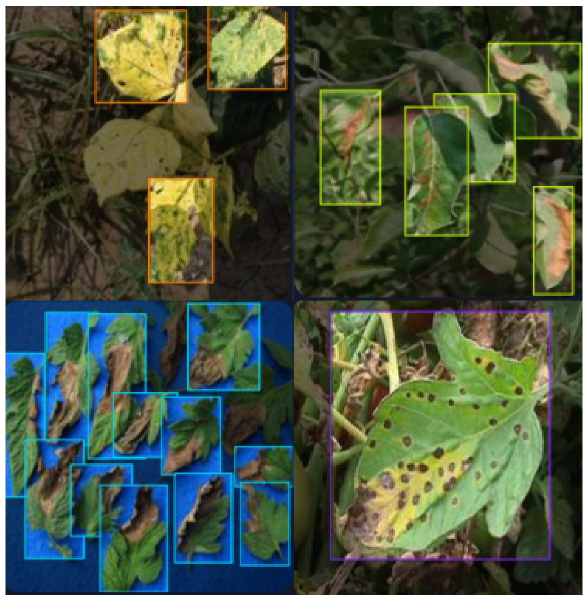

Figure 3-4: Apple dataset samples

3.3.2 Model Architecture and Training

In this research, a ResNet50 model was used as the backbone for the disease classification task. The ResNet50 model was pre-trained with weights from ImageNet but without top classification layers. This was to take advantage of the deep feature extraction abilities of ResNet50 while providing task-specific flexibility. ResNet50 layers were frozen in order to retain the knowledge accumulated from ImageNet at early training stages, whereby the model would be able to focus on learning task-related patterns without relearning low-level features.

Various strategies were employed in the network design to avoid overfitting and generalize the model. Global Average Pooling (GAP) was applied on top of the ResNet50 backbone output to downsample the spatial dimensions and pool the features in a way that the feature representation is compact. Then, a fully connected layer with 512 neurons and ReLU activation was appended to learn the high-level relations among the data. Next, a Dropout layer with the rate of 0.5 was added to extra-regular the model and prevent overfitting by randomly setting a fraction of the input units to zero to zero during training time. Finally, the output layer used the softmax activation function to produce one of the five disease class predictions for the given input image.

Training was performed on the model with the Adam optimizer and a learning rate of 1e-5 to ensure stable and effective training. The decreased learning rate helped fine-tune the model without disturbing the pre-trained ResNet50 features. Categorical cross-entropy was the loss function used, which is applicable in multi-class classification. Additionally, the model was also tested with accuracy, precision, recall, and AUC to completely understand its performance.

The model was trained on a train-validation split for 100 epochs. Validation data were used to monitor the generalization capability of the model as training progressed, thus minimizing overfitting.

3.4 Integrated Models

3.4.1 Dataset Preparation and Preprocessing

To evaluate the performance of the integrated model, a dataset consisting of 50 images was created. All the images have apple trees with apples and leaves visible in them. Images were selected particularly to test whether the model could detect and classify more than one component in the same image. All the 50 images were used only for testing the effectiveness of the model. Prior to processing, images were resized from their original size to a size of 640×640 pixels in order to align with YOLOv5 model's input requirements.

Figure 3-5: Tree dataset samples

3.4.2 Model Architecture

After training and validating the individual models, they were merged into one pipeline for end-to-end apple tree health diagnosis. The integrated system begins by loading the three main models: YOLOv5 for object detection, EfficientNet for classifying apple quality, and ResNet50 for diagnosing leaf diseases. The YOLOv5 model, which had been fine-tuned for detecting apples and leaves, was used to perform object detection on input orchard images. Detected apple and leaf crops were then kept in separate folders for further analysis.

For apples, cropped images were initially preprocessed by resizing to 224×224 pixel size and EfficientNet specification normalization. These images were inputted to the EfficientNetB0 model for predicting whether the apple was rotten or not. Prediction decoding was carried out based on comparing output probabilities for classes. Leaf crop images were also resized and preprocessed for being passed to ResNet50. They were then passed through the trained ResNet model, which produced results as normal or one of the five disease classes: Early blight, Late blight, Leaf spot, Mosaic virus, or Rust. Class labels were then added to the model outputs in order to provide a general indication of likely leaf diseases. The overall solution enables the system to automatically segment apples and leaves, assess apple quality, and diagnose leaf diseases from a single image input.

Chapter 4

Results

An analysis of the test performance of three models, YOLOv5, ResNet and EfficientNet, evaluated on separate datasets. YOLOv5 was tested on merged apple dataset and leaf dataset, EfficientNet was tested and trained on the apple dataset, while ResNet was used in leaf classification. The accuracy of all models were measured based on how they could distinguish between different classes of their respective datasets.

4.1 YOLOv5 performance on Object Detection

The YOLOv5 model was evaluated on a test dataset comprising 559 images and a total of 1,115 object instances. As shown in Table 4.1, the model achieved good overall performance, with a precision of 0.909 and a recall of 0.690 across all classes. The mean average precision at IoU threshold 0.5 (mAP50) reached 0.781, while the mAP50-95 was 0.609, reflecting the model's solid detection capabilities across varying IoU thresholds.

Class	Instances	Precision (P)	Recall (R)	mAP50	mAP50-95
all	1115	0.909	0.690	0.781	0.609
leaf	816	0.874	0.575	0.667	0.520
apple	299	0.945	0.805	0.895	0.698

Table 4.1: Evaluation results for the YOLOv5 model on the test set.

Apple objects were detected with the highest accuracy, achieving a precision of

0.945 and recall of 0.805. The leaf class, while still demonstrating good precision (0.874), had a lower recall (0.575), suggesting a loss of some leaf instances when detecting. This disparity indicates the need for further improvements in detecting smaller or less distinct leaf features.

On average, the YOLOv5 model behaved well for detecting and localizing apples and leaves with high precision and similar values of mAP, confirming its suitability for object detection real-time applications in the agricultural context.

4.2 EfficientNet Performance on Apple Classification

EfficientNetB0 was trained from scratch without pre-loaded ImageNet weights, allowing the model to learn all feature representations specifically from the apple classification dataset. The model was adapted to a binary classification task that differentiates between fresh and rotten apples. A custom head was appended to the base EfficientNet architecture, including global average pooling, a dense hidden layer with L2 regularization, and a softmax output layer for final prediction.

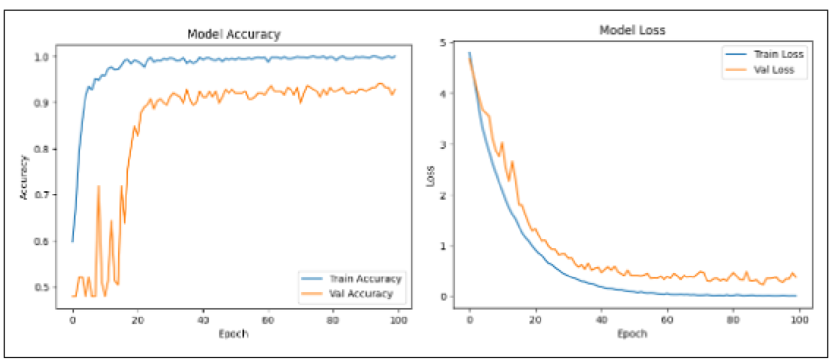

Figure 4-1: EfficientNet accuracy and loss on training

After training for 100 epochs, the model demonstrated strong performance on the unseen test dataset and achieved a test accuracy of 0.9412, which is proof of good performance. Other result metrics are shown in Table 4.3.

These results highlight EfficientNet's capacity to learn discriminative features in the apple health classification problem, including fine texture, color, and size differences. The Area Under the Curve (AUC) value confirms the model's ability to

Metric	Value
Accuracy	0.9412
Precision	0.9412
Recall	0.9412
AUC	0.9857
Loss	0.1908

Table 4.2: EfficientNetB0 evaluation metrics on the apple classification test set.

discriminate between the two classes with high accuracy at different thresholds, and the balanced precision and recall demonstrate a good-generalized classifier with minimal bias towards either class.

However, although it did well on the test, the model performed worse when tested on custom images, particularly those in which apples were not zoomed in separately or had more than one apple per frame. The poor performance reflects weak generalization outside the curated dataset environment. A potential explanation is that training and testing sets were largely populated with pictures having apples separated and centered, and it became easier to classify and extract features from them. By contrast, authentic images of occluded backgrounds, different sizes, and more than one object made it challenging to localize and classify apple conditions of the model with reliability. These findings emphasize the importance of dataset diversity and image realism during training while looking for well-generalizing models in field environments.

4.3 ResNet Performance on Leaf Classification

ResNet was applied in leaf classification with its residual connections to extract deep features. The model's performance when tested on the test dataset was an accuracy of 0.7094, which indicates moderate classification capability. This is a reflection that ResNet was having trouble differentiating sick from healthy leaves with differences in leaf texture, signs of diseases, and background data set noises. While the precision of 78.% is good for minimizing false positives, the recall of 64.10% indicates a potential area for improvement. With an AUC of 0.9047, the model demonstrated strong discriminatory power between classes.

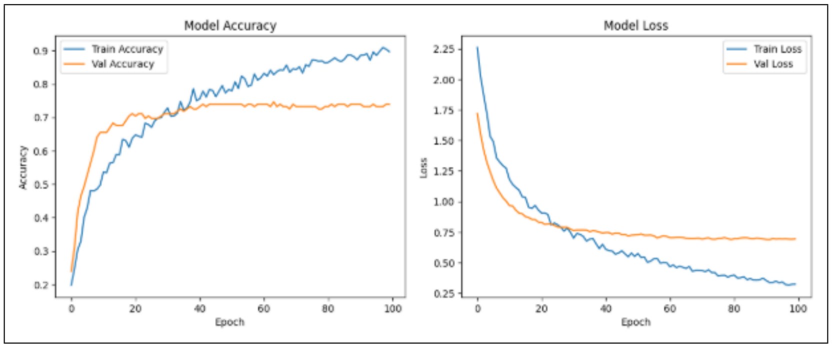

Figure 4-2: Resnet accuracy and loss on training

Metric	Value
Accuracy	0.7094
Precision	0.7812
Recall	0.6410
AUC	0.9047
Loss	0.8833

Table 4.3: ResNet evaluation metrics on the leaf classification test set.

While ResNet has the capacity to learn hierarchical representations, its performance on leaf classification as a leaf suggests inadequacies in learning subtle distinctions among leaf conditions. The complexity of the dataset, including lighting and superimposed visual patterns, may have weakened the model's generalizability. While ResNet is a resilient baseline model, its performance suggests that improvements can be made with greater data augmentation, more advanced preprocessing techniques, or employing different architectures that are better suited for fine-grained leaf classification.

Overall, the individual dataset comparison of ResNet and EfficientNet reveals the influence of model structure on the accuracy of classification. Despite the fact that EfficientNet emerged superior in the classification accuracy of the apple, the underperformance of ResNet on the leaf classification signifies the possibility of having a specialized strategy for identifying the leaf condition.

4.4 Integrated Model Performance

The integrated model, which combines YOLOv5 for object detection, EfficientNet for assessing apple quality, and ResNet for evaluating leaf quality, was tested on a dataset comprising 50 apple tree images where 30 categorized as healthy and 20 as diseased. The YOLOv5 component was responsible for identifying individual apples and leaves within each image, serving as the initial step in the quality analysis pipeline. Results are shown in Table 4.4.

Dataset	Total Number of Detected Objects	Predicted Correctly	Predicted Incorrectly	Accuracy (%)
Healthy tree	115	93	22	80.9%
Diseased tree	42	36	6	85.72%
Overall	157	129	28	82.1%

Table 4.4: Accuracy information for healthy and diseased apple images.

From the healthy tree images, a total of 115 objects (apples and leaves) were detected. Of these, 93 objects were correctly classified by the subsequent EfficientNet and ResNet models, while 22 were incorrectly classified, resulting in an overall classification accuracy of 80.9% for healthy trees. Although the results are good, there are several reasons for incorrect identification. Among the most significant was the presence of intense sunlight in a number of the healthy tree images as shown in figure 4-3. Increased sun ray exposure on apples and leaves led to glare and reflections, which in turn created inconsistency in object appearance and color deceiving the model during the classification stage. Moreover, healthy trees have dense foliage and overlapping fruits, creating additional visual clutter that interferes with correct detection and assessment.

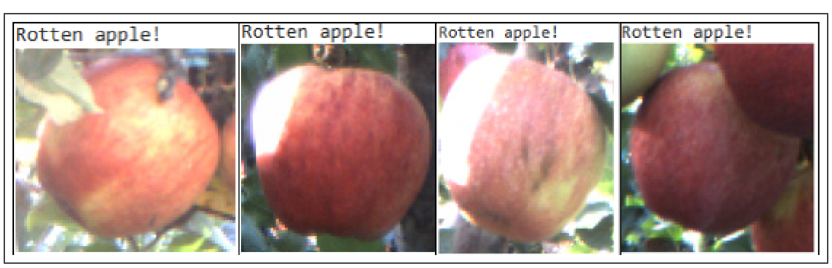

Figure 4-3: Examples of bad object capture

On the other hand, the images of diseased trees provided 42 objects detected.

Of these, 36 were correctly classified and 6 were misclassified, which gave a slightly improved accuracy of 85.7%. Diseased trees contain less foliage and fewer fruits, which reduce occlusion and glare, and improve the model to detect and classify objects more consistently.

Overall, 157 objects were detected across all images, of which 129 were correctly identified and 28 were misclassified, resulting in a total accuracy of 82.1%. From the results, it is clear that the combined model is shows appropriate accuracy in analyzing tree conditions, but also requires further optimization in dealing with lighting variation and complex backgrounds to improve accuracy.

Chapter 5

Conclusion

This thesis addresses challenges in apple orchard management by developing a diagnostic system that uses machine learning to assess the health of individual trees through image analysis. By using YOLOv5 for object detection, EfficientNetB0 for classifying the quality of apples, and ResNet50 for leaf disease detection, the system aims to provide accurate, automatic diagnoses of apple tree health. Experimental outcomes verify that every model attains good performance in its individualized task, and their combination enables a comprehensive evaluation pipeline. Despite of the current issues such as lighting variation and occlusion background, the system holds the promise of good performance capability for real-time tracking and facilitation of decision-making in orchard environments. This approach enables the more efficient, sustainable, and information-driven orchard management that reduces reliance on visual inspections and enables timely actions to maintain production and fruit quality.

Bibliography

[1] B. Kiczorowska and P. Kiczorowski. Comparison of some chemical components in apples szampion variety produced in the lublin province. *Acta Scientiarum Polonorum Technologia Alimentaria*, 4(2):37–45, 2005.

[2] C. Feng, H. R. Wu, H. J. Zhu, and X. Sun. The design and realization of apple orchard intelligent monitoring system based on internet of things technology. *Advanced Materials Research*, 546:898–902, 2012.

[3] B. Yan, P. Fan, X. Lei, Z. Liu, and F. Yang. A real-time apple targets detection method for picking robot based on improved yolov5. *Remote Sensing*, 13(9):1619, 2021.

[4] Zhijun Li, Shenghui Yang, Deshuai Shi, Xingxing Liu, and Yongjun Zheng. Yield estimation method of apple tree based on improved lightweight yolov5. *Smart Agriculture*, 3(2):100, 2021.

[5] G. Xuan, C. Gao, Y. Shao, M. Zhang, Y. Wang, J. Zhong, and H. Peng. Apple detection in natural environment using deep learning algorithms. *IEEE Access*, 8:216772–216780, 2020.

[6] J. Wang, Y. Su, J. Yao, M. Liu, Y. Du, X. Wu, and M. Zhao. Apple rapid recognition and processing method based on an improved version of yolov5. *Ecological Informatics*, 77:102196, 2023.

[7] L. A. Aldakhil and A. A. Almutairi. Multi-fruit classification and grading using a same-domain transfer learning approach. *IEEE Access*, 2024.

[8] M. A. C. Dionisio, I. J. D. Salazar, and C. C. Hortinela. Efficientnet-lite 4-based classification system for grading philippine strawberries. In *2024 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*, pages 366–371. IEEE, 2024.

[9] J. Wen and J. He. Agricultural development driven by the digital economy: improved efficientnet vegetable quality grading. *Frontiers in Sustainable Food Systems*, 8:1310042, 2024.

[10] R. C. D. B. Nigam. Enhancing automated agricultural product grading with advanced deep learning techniques. *Machine Intelligence Research*, 18(1):1138–1148, 2024.

[11] N. Ismail and O. A. Malik. Real-time visual inspection system for grading fruits using computer vision and deep learning techniques. *Information Processing in Agriculture*, 9(1):24–37, 2022.

[12] X. Chao, G. Sun, H. Zhao, M. Li, and D. He. Identification of apple tree leaf diseases based on deep learning models. *Symmetry*, 12(7):1065, 2020.

[13] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang. Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. *IEEE Access*, 7:59069–59080, 2019.

[14] X. Li and L. Rai. Apple leaf disease identification and classification using resnet models. In *2020 IEEE 3rd International Conference on Electronic Information and Communication Technology (ICEICT)*, pages 738–742, 2020.

[15] V. Kumar, H. Arora, and J. Sisodia. Resnet-based approach for detection and classification of plant leaf diseases. In *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, pages 495–502, 2020.

[16] P. Sirenjeevi, J. M. Karthick, K. Agalya, R. Srikanth, T. Elangovan, and R. Nareshkumar. Leaf disease identification using resnet. In *2023 International*

Conference on Artificial Intelligence and Knowledge Discovery in Concurrent Engineering (ICECONF), pages 1–5, 2023.

[17] X. Zhang, H. Li, S. Sun, W. Zhang, F. Shi, R. Zhang, and Q. Liu. Classification and identification of apple leaf diseases and insect pests based on improved resnet-50 model. *Horticulturae*, 9(9):1046, 2023.

[18] Plants. Healthy and sick dataset [open source dataset]. `https://universe.roboflow.com/plants-g079u/healthy-and-sick-s1cqh`, January 2025. Retrieved March 20, 2025.

[19] DeluxeViper. Fridge ingredients dataset. `https://universe.roboflow.com/deluxeviper/fridge-ingredients`, 2023. Visited on 2025-04-10.

[20] Ennur. Apple type dataset [open source dataset]. `https://universe.roboflow.com/ennur/apple-type`, December 2023. Retrieved March 20, 2025.

[21] NataliiaLevk. Plant disease detection dataset [open source dataset]. `https://universe.roboflow.com/nataliialevk/plant-disease-detection-qidwz`, January 2025. Retrieved March 20, 2025.