Innovative methodologies in agriculture for high-throughput plant phenomics using computer vision and artificial intelligence

Department of Electrical and Information Engineering

INDUSTRY 4.0

Ph.D. Program

SSD: ING-INF/05- INFORMATION PROCESSING SYSTEMS

Final Dissertation

Innovative Methodologies in Agriculture for High Throughput Plant Phenomics Using Computer Vision and Artificial Intelligence

by

Firozeh Solimani

Supervisors:

Dr. Vito Reno`

Prof. Giovanni Dimauro

Coordinator of Ph.D. Program:

Prof. Caterina Ciminelli

Course n°37, 01/11/2021-31/10/2024

Department of Electrical and Information Engineering
INDUSTRY 4.0
Ph.D. Program
SSD: ING-INF/05- INFORMATION PROCESSING SYSTEMS

Final Dissertation

Innovative Methodologies in Agriculture for High Throughput Plant Phenomics Using Computer Vision and Artificial Intelligence

by
Firozeh Solimani

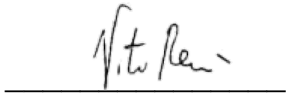
Referees:

Prof. Carmela Rosaria Guadagno

Prof. André Pierre Marie Fabbri

Supervisors:

Dr. Vito Reno`



Prof. Giovanni Dimauro

Coordinator of Ph.D Program:
Prof. Caterina Ciminelli

Course n°37, 01/11/2021-31/10/2024

To bright and self-aware souls who make the path of life clear for everyone

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Dr. Vito Renò, for his exceptional guidance, mentorship, and support throughout my PhD journey. From the beginning, Dr. Renò has been a constant source of inspiration, wisdom, and encouragement. I am also grateful to my second supervisor, Dr. Prof. Giovanni Dimauro, for his valuable input and suggestions during the development of this work. Insightful advice and guidance from these two honourable supervisors have been invaluable in shaping my research and in helping me to achieve my goals.

I am grateful to my esteemed colleague, Dr. Angelo Cardellicchio, at the Institute of Intelligent Industrial Technologies and Systems for Advanced Manufacturing, National Research Council of Italy, for his invaluable assistance and cooperation. His guidance and support have significantly contributed to my academic progress.

I am grateful to Dr. Angelo Petrozza, Dr. Stephan Summerer and Dr. Francesco Cellini at the ALSIA Centro Ricerche Metapontum Agrobios-Italy, for providing access to a diverse range of images that were integral to this research. Their plant phenomics platform facilitated the acquisition of high-quality visual content, enriching the presentation and analysis within this thesis.

This thesis would not have been possible without the collective support and encouragement of all those mentioned above. Thank you for being a part of this transformative journey.

Abstract

Plant phenotyping is essential for plant breeding and crop management, but traditional methods are labor-intensive and prone to errors. Computer vision and deep learning (DL) offer solutions by rapidly and accurately analyzing plant images. However, data management, annotation, and preprocessing for deep learning models can be costly and time-consuming. Additionally, advanced models may require architectural modifications to minimize computational costs, streamline the models, and enhance their performance for optimal diagnostics.

This study seeks to systematically review the key hardware and software elements that influence high-throughput plant phenotyping. It will also delve deeply into the software and algorithms used in this field. The research will particularly emphasize innovative methodologies in data management and pinpoint the most effective algorithms for analyzing data generated by plant phenotyping platforms using computer vision and artificial intelligence within a laboratory setting. A deep learning model (YOLOv5) be designed to effectively recognize diverse morphological features across various plant species. This model, coupled with transfer learning and rigorous evaluation techniques, achieved notably high scores in precision, recall, and F1-measure, adeptly addressing the unique challenges posed by the input images.

This research introduces an innovative approach to address dataset imbalances using data balancing techniques. By pooling the data and generating extra samples for underrepresented classes, the dataset is rebalanced. Moreover, an attention module is integrated into the proposed head model architecture (YOLOv8) to enhance the detection capability of target classes. These methods enable the training of deep learning models with significantly improved accuracy. The optimization of model head adaptations to enhance the detection of small objects was presented, utilizing the basic architecture of YOLOv8. As a result, the integrated SO-YOLOv5 model demonstrates higher accuracy in detecting small objects while minimizing

computational costs and maintaining simplicity. An alternative approach to the RSA segmentation problem was presented, employing binary classification through probabilistic map estimation to classify original image pixels as background or foreground. This work introduces a comprehensive processing pipeline for end-to-end analysis of factory RSAs.

In conclusion, this thesis develops methods to reduce time and effort and also increase accuracy and performance for applying DL models in plant phenotyping. It investigates single-stage detectors that can detect aerial parts of plants, and a comprehensive processing pipeline for end-to-end analysis of factory RSAs by CNNs models. By making DL models more accessible and scalable, this research advances plant phenotyping and crop production.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The escalation of greenhouse gas emissions globally stems from the exponential rise in human population alongside heightened consumption of goods and services, intensifying the ramifications of climate change. Consequently, the severity and recurrence of extreme weather phenomena such as heatwaves, droughts, storms, and floods have surged [5]. These occurrences not only imperil food security but also pose substantial risks of widespread malnutrition. The exponential growth of the global population in recent decades, which surged from 1.8 billion in 1915 to 7.7 billion in 2019, and is projected to reach approximately 10 billion by 2050 [150], has significantly heightened the demand for sustainable food supply. However, on the one hand, the adverse consequences arising from disturbances in plant growth, climate variability, land use changes, susceptibility, resilience, structural composition, physiological functions, and environmental dynamics have reduced crop yields, resulting in setbacks to agricultural production. On the other hand, extensive agricultural practices have intensified the environmental crisis, leading to the depletion of natural resources. One important concern is the widespread occurrence of drought affecting a considerable portion of the global population [189]. Drought has the potential to disrupt plant production, causing changes in nutrient absorption and overall plant functionality [42].To mitigate these challenges, the strategic use of biological stimulants during key plant growth stages has shown potential in restoring productivity [23].

1.1 Plant Phenotyping in Agriculture

Research-based solutions aim to expedite plant breeding by crafting genotypes adept at adapting to abiotic stresses, potentially boosting crop yields[157]. Researchers believe that employing biological stimulants during key plant growth stages can boost plant productivity [37],[172]. Moreover, they state that progress and widespread integration of advanced sensing technologies has heralded a new era in monitoring unpredictable agricultural ecosystems [200] through the development of optimized smart farming procedures [195] and high-throughput phenotyping techniques [137]. Precision agriculture has emerged as a promising strategy for enhancing production, improving both the quality and quantity of yields, protecting the environment, and reducing costs [64]. However, the success of these systems depends heavily on the availability of high-quality data [130]. But despite the strides made in generating new lines with enhanced adaptability to specific pedoclimatic factors, the process remains hampered by phenotyping, and constraining breeding programs [6]. One avenue that has proven effective in generating such data is the use of genomic tools, which can provide detailed insights into the genetic composition of plants [199]. While genomic characterization is a critical first step, the systematic quantification of phenotypic traits remains a significant challenge [31]. This has led to the growth of the field of plant phenotyping, which, when integrated with genomic data, offers a powerful tool for advancing plant breeding and improving crop performance [158]. Figure 1.1 illustrates how the interaction between genotype and environment results in the expression of observable traits, known as the phenotype.

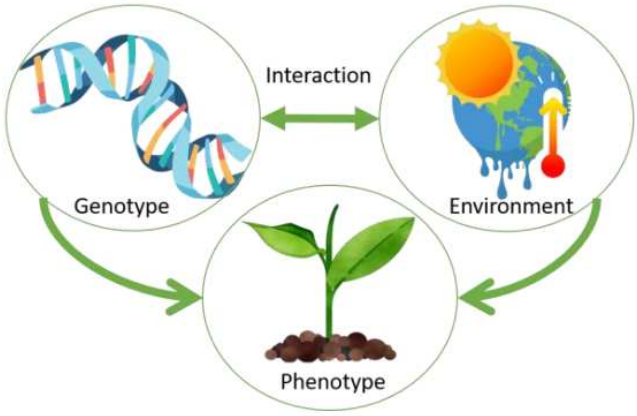


Fig. 1.1 The interaction between genotype and environment leads to the expression of observable traits, which are referred to as the phenotype [103]

In various agricultural contexts, characterizing the physical attributes of plants, known as phenotyping [152], is an inevitable and necessary thing. This indicates that plant breeders depend on phenotypic data to assess crop performance [211],[6]. Phenotyping, formally defined in [152], entails evaluating plants performance for desired traits. Plant phenotyping focuses on the assessment of both morphological and physiological characteristics of plants. Morphological traits encompass features such as leaves, stems, flowers, fruits, and roots, providing essential information about the vegetative and reproductive status of the plant[168]. Physiological traits, including vigor, leaf surface, biomass, and inflorescence structure, are used to assess the overall health and performance of plants throughout their life cycle [202]. Accurate measurement of these traits is critical for improving crop monitoring, assessing plant stress levels, and informing the development of more efficient agricultural practices [6]. For example, specific phenotypic traits like leaf area have been found to correlate with above-ground biomass [151]. Furthermore, agronomists employ plant spacing and density as predictive indicators for the future yield of their crops [35]. By synergizing high-throughput phenotyping techniques with genotypic data, scientists have enhanced the scrutiny of crop yield, its intricate components, and quality attributes, while concurrently appraising resistance against both abiotic and biotic stresses [30], [58]. Despite the importance of plant phenotyping, traditional methods have been constrained by high costs, time consumption, and the potential for destructive testing [36]. Traditional phenotyping methods, which rely heavily on manual observations and basic measurement tools, are often insufficient to meet the precision and efficiency required for large-scale breeding programs. These methods tend to lack the comprehensive accuracy necessary for assessing diverse traits across vast populations of plants, especially under varying environmental conditions. Consequently, they impede the identification of superior genotypes that exhibit enhanced stress tolerance traits essential for developing resilient crop varieties that can thrive in challenging environments [56]. Stress factors like drought, salinity, and extreme temperatures interact with plant physiology in ways that are not always straightforward. These responses often involve multiple genes and regulatory pathways, making it difficult to assess the full scope of a plant resilience using traditional phenotyping methods. These approaches typically fall short when it comes to nuanced phenotypic evaluations, as they cannot capture the dynamic responses plants exhibit in fluctuating environmental conditions [240]. As a result, breeding programs face significant challenges when attempting to develop crop varieties capable of withstanding the

adverse effects of climate change, ultimately delaying progress in enhancing global food security. Despite encountering challenges such as cost, performance limitations, and incomplete coverage, there is a pressing need to enhance current methodologies for measuring phenotypic traits by advancing automation throughout the entire phenotyping process, from data acquisition to interpretation. Modern phenotyping technologies such as high-throughput phenotyping (HTP) systems, remote sensing, and machine learning algorithms are proving to be essential tools in addressing the limitations of traditional methods. These technologies enable breeders to capture large volumes of data quickly and accurately, under both controlled environments and field conditions, allowing for more precise evaluation of phenotypic traits across large populations of plants [36].

1.2 Hardware tools in Plant Phenotyping

New hardware tools have been developed to enhance the throughput of plant phenotyping, giving rise to the field of high-throughput phenotyping (HTP) [56]. HTP utilizes automated platforms to extract plant traits in a non-destructive manner, significantly reducing the time and labor required for phenotypic analysis [6]. These platforms are equipped with advanced sensors and technologies, including unmanned vehicles, drones, and satellites, which allow for the large-scale collection of phenotypic data [223]. The choice of platform and sensor type depends on the specific application scenario. For example, aerial platforms are generally more flexible and efficient for monitoring plants in open fields, while ground-based robotic platforms are more suited to controlled environments [12]. Satellite-based platforms are particularly useful for collecting data over large geographic areas [14]. Although HTP has the potential to revolutionize plant phenotyping, it also generates vast amounts of data that require efficient processing and analysis [184]. The integration of biological expertise with computer science and engineering is essential for managing these large datasets. In particular, non-invasive imaging techniques have become increasingly important in phenotypic data collection, with sensors such as visible RGB cameras, near-infrared (NIR) sensors, and hyperspectral imaging systems playing a central role in capturing critical plant traits [107]. RGB sensors, for instance, are widely used to capture images of plant morphology, while NIR sensors assess important physiological parameters such as chlorophyll content, water levels, and temperature

[227]. Hyperspectral imaging allows for the detailed investigation of leaf tissue structure and pigment composition [126]. In addition to these imaging systems, advanced techniques such as multi-spectroscopy, Raman spectroscopy, magnetic resonance imaging (MRI), and X-ray imaging have been employed to study plant internal anatomy, including the structure of fruits, seeds, and roots [142]. X-ray imaging, in particular, has proven valuable for studying root system architectures, which are difficult to analyze due to their interaction with surrounding soil [47]. Specialized tools like rhizotubes and minirhizotrons, which are buried underground, allow for the high-resolution imaging of root systems without disturbing the soil environment [118].

1.3 Data and Artificial Intelligence in Plant Phenotyping

As discussed by [184], while HTP holds the potential to revolutionize plant phenotyping by improving the precision and scale of data collection, the sheer volume of data produced—especially when using non-invasive imaging techniques and sensor arrays—requires advanced computational techniques to make sense of the data. In addition to the computational challenges, there is also the issue of data storage and sharing. Large HTP datasets require significant storage capacity, often distributed across cloud or high-performance computing infrastructures. This requires collaboration between biologists and IT specialists to ensure the datasets are properly managed, secured, and made accessible to a wider scientific community. Moreover, developing user-friendly tools for visualization and interpretation of HTP data is critical, allowing plant scientists to interact with the data in intuitive ways and apply their biological expertise effectively. This requires collaboration between biologists and IT specialists to ensure the datasets are properly managed, secured, and made accessible to a wider scientific community. For example, as [184] highlights, machine learning (ML) algorithms can be used to detect patterns and extract phenotypic traits from high-dimensional datasets. However, once collected, the vast amount of phenotypic data must be labeled, typically by domain experts, before it can be analyzed using machine learning (ML) and computer vision techniques [136]. ML algorithms, such as support vector machines (SVMs), have been successfully applied to classify plant traits, with studies showing improvements in classification accu-

racy when ML is combined with computer vision [196]. For instance, ML-based approaches have been used to segment plant images and classify leaves with high accuracy, streamlining the phenotyping process [160]. However, to make ML effective in this context, researchers must also implement robust data-cleaning, annotation, and feature-extraction workflows. However, ML approaches face limitations when dealing with low-quality images or varying imaging conditions. Deep learning (DL) models have emerged as a more robust alternative, as they can automatically learn features from input data and generalize well across diverse datasets [90]. DL models, however, require large amounts of training data to perform effectively [101]. Both ML and DL approaches have strengths in plant phenotyping; while ML excels in regression tasks, DL has shown superior performance in classification tasks due to its ability to quickly and accurately identify plant features [185]. Recent strides in the fields of CV, ML, and DL have enabled the development of non-invasive imaging techniques that can capture vast amounts of plant phenomics data with unprecedented precision and speed. This technological progress is vital for automating the phenotyping process, enabling the large-scale, high-throughput analysis of plant traits under both controlled and field conditions. The ability to non-invasively monitor plant growth and stress responses over time provides researchers with detailed phenotypic information that can be used to make informed decisions in crop breeding and management [196]. Despite these advancements, there is a need for more comprehensive studies that integrate various, including sensor technologies, data processing methods, and imaging techniques. Addressing these complexities will be key to improving the accuracy and efficiency of plant phenotyping, which in turn will contribute to the development of more sustainable and productive agricultural systems [95]. Automation, standardization, and the incorporation of quantitative analysis techniques into the phenotyping process will be crucial for maximizing the potential of plant phenomics. The continued development and refinement of machine learning and AI tools, along with greater investments in infrastructure and expertise, will help overcome the current limitations of phenotyping systems, ultimately leading to more efficient and accurate identification of superior genotypes.

1.4 Computer vision-based plant phenotyping

Experimental designs for plant phenotyping often leverage computer vision techniques to enable precise quantitative measurements. These methods facilitate the analysis of gene-environment interactions, plant growth infrastructure, substrate management, and various monitoring systems. Developing standardized protocols for plant monitoring requires imaging sensors to capture and process accurate imaging data. Additionally, metadata plays a critical role in evaluating phenotype data, enabling informed decision-making to optimize growing environments, whether in fields, greenhouses, or controlled growth chambers. Various plant imaging techniques can support these objectives, and the following sections explore some of the available datasets. Recent advancements in computer vision systems have significantly enhanced both hardware and software components. The hardware—comprising cameras, lighting systems, and communication devices—serves as the foundation, while software, including image processing algorithms, forms the system's core. A reliable image acquisition setup depends heavily on suitable illumination, which can include point, strip, ring, backlight, structured, or combined light sources. These can be further classified into LED, halogen, and high-frequency fluorescent light sources. Cameras used in these systems may feature either global or rolling shutters, each suited to specific imaging requirements.

1.4.1 Pre-processing

Pre-processing plays a crucial role in image processing, as it refines the region of interest, resulting in more accurate segmentation or classification. Additionally, it can reveal fine details, such as tiny grains or spiked particles in plants. Data augmentation techniques like cropping, rotation, and scaling are also commonly employed to introduce variations that make models more robust and capable of generalizing patterns. Noise reduction and enhancement methods, widely used in image processing, are particularly beneficial in removing noise from plant imaging data. For example, techniques such as the fuzzy operator and morphological operator methods are effective in addressing salt-and-pepper noise, leveraging processes like erosion and dilation. A morphological dual operator approach has also been used for noise removal, with performance validated through metrics like the peak signal-to-noise ratio (PSNR). Additionally, hybrid methods combining nonlinear filters have

shown success in eliminating salt-and-pepper noise. For monochromatic images, a two-phase noise reduction strategy has been developed. This approach involves detecting noise in the first phase and removing it with adaptive filtering in the second phase. Evaluations using PSNR values have demonstrated the reliability of this method in reducing noise effectively [131]. Overall, pre-processing techniques in image processing are essential for enhancing the region of interest by removing noise, improving contrast, and addressing artifacts. These methods provide a way to correct imperfections in plant phenotyping images, ensuring higher-quality data for downstream analysis.

1.4.2 Deep Learning for Computer Vision

Since 2012, convolutional neural networks (CNNs) have become the leading approach for computer vision tasks, consistently outperforming traditional machine learning methods [45]. CNNs are deep learning frameworks designed for image processing and recognition, capable of automatically learning features from input data. Through training and parameter optimization, CNNs apply multiple layers of nonlinear transformations to the input, progressively combining low-level features into high-level semantic representations. Unlike conventional machine learning methods, CNNs leverage deep architectures that enable more efficient training and reduce the need for manual feature engineering, streamlining the data processing workflow.

Convolutional Neural Networks (CNNs) are a specialized class of neural networks designed to utilize the spatial relationships within input data. Initially proposed by [59], CNNs struggled to gain traction due to the limited computational power available at the time for training. However, in the 1990s, [102] introduced a gradient-based learning approach for CNNs, achieving notable success in the task of handwritten digit classification. Since then, CNNs have become a cornerstone of computer vision, finding applications in areas such as facial recognition, object detection, robotics, and autonomous vehicles.

A typical CNN architecture consists of a series of convolutional and pooling layers, with pooling layers typically following convolutional ones. The final stages of the network include fully connected layers, and the last layer is typically a softmax classifier, as shown in Figure 1.2. Each layer in the network processes the input data,

transforming it into different representations, eventually mapping the input to a 1D feature vector in the fully connected layers.

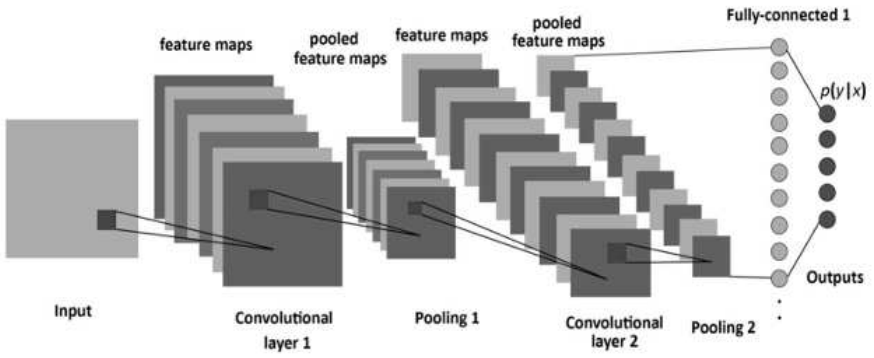


Fig. 1.2 The structure of a CNN, consisting of convolutional, pooling, and fullyconnected layers.[4]

The core components of a CNN can be broken down into three key types of layers: (i) convolutional layers, (ii) pooling layers, and (iii) fully connected layers, each serving a distinct function.

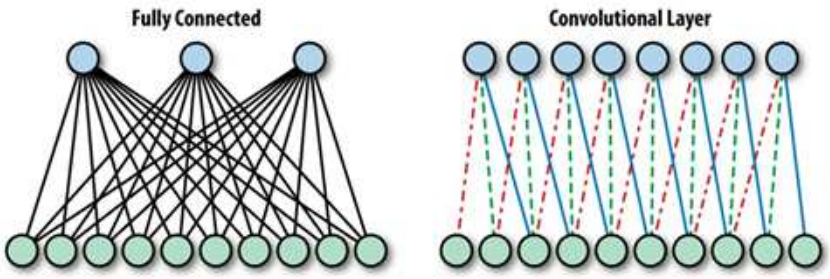


Fig. 1.3 In a fully connected layer (left), each unit is connected to all units of the previous layers. In a convolutional layer (right), each unit is connected to a constant number of units in a local region of the previous layer. The figure and description are taken from [110]

- Convolutional Layers: These layers perform convolutions on the input image or feature maps using different kernels, generating various feature maps. The convolution operation is advantageous for learning spatial hierarchies of features, and several studies have proposed replacing fully connected layers with convolutions to speed up the learning process. A comparison of fully connected and convolutional layers is illustrated in Figure 1.3.

- Pooling Layers: Pooling layers are responsible for reducing the spatial dimensions of the feature maps produced by the convolutional layers. This downsampling operation does not affect the depth of the data but reduces the size of the feature maps, thereby decreasing computational complexity and helping prevent overfitting. Common pooling strategies include max pooling and average pooling. Research has demonstrated that max pooling can accelerate convergence, improve generalization, and select more robust features [19],[178].
- Fully Connected Layers: After the convolutional and pooling layers, fully connected layers perform high-level reasoning by connecting all neurons in the previous layers. These layers transform the 2D feature maps into a 1D feature vector, which can be used for classification or further processing, depending on the task at hand.

In this context, this dissertation presents a comprehensive and systematic review of the existing methodologies employed in the field of plant phenotyping. This review not only examines the traditional techniques currently in use but also aims to identify and explore novel methodologies in data management that can enhance the efficiency and effectiveness of data handling processes. These techniques encompass a spectrum that includes accurate and rapid detection and monitoring of plant traits, specifically the identification of morphological traits such as fruits, flowers, nodes, and roots, as well as the assessment of model accuracy and performance in trait detection. Additionally, the modification of model architectures was studied to enhance recognition accuracy and reduce computational costs associated with the base model weights.

1.5 Research Gaps and Objectives

1.5.1 Deep Learning Techniques Applied to Plant Stress (Biotic and Abiotic) Phenotyping

Deep learning (DL) techniques have emerged as transformative tools in plant stress phenotyping, addressing both biotic (e.g., diseases, pests) and abiotic (e.g., drought, salinity) stresses. These methods leverage large datasets, such as high-resolution

images, to detect subtle stress indicators that traditional methods often overlook. DL models, particularly convolutional neural networks (CNNs), have demonstrated remarkable accuracy in classifying plant stresses. For example, CNNs have been successfully applied to identify and classify various foliar stresses in soybean plants, achieving high levels of precision in differentiating between stress types [183]. In other hand, DL enables the rapid processing of extensive datasets, facilitating large-scale phenotyping essential for modern agricultural practices. The integration of DL with high-throughput phenotyping platforms has proven critical for evaluating plant health and stress responses efficiently [109].An interesting approach is temporal analysis of image data using DL models allows for early detection of plant stress, which is crucial for timely intervention and mitigation of yield losses. DL approaches have been effectively utilized to monitor water-deficiency-induced stress in plants, offering early warning systems for precision agriculture [234]. [32] highlight that the combination of deep learning (DL) with hyperspectral and thermal imaging significantly enhances the detection and quantification of plant stresses. They emphasize that this integration offers a comprehensive understanding of plant health by capturing both physiological and biochemical changes associated with various stress condition.

Despite significant advancements, several challenges persist in the application of deep learning (DL) to plant stress phenotyping. One major limitation is the reliance on large, annotated datasets, which are often time-consuming and resource-intensive to produce. Developing standardized datasets and leveraging advanced data collection techniques could significantly reduce the barriers to progress, facilitating more accurate and scalable research across the field, and also use of novel strategies to circumvent the need for accurately labeled data for training the DL tools. Another approach is the development and implementation of innovative techniques that minimize or eliminate the reliance on large, accurately labeled datasets for training deep learning (DL) models. Circumventing the need for accurately labeled data involves finding ways to train deep learning (DL) models effectively without relying on large, perfectly labeled datasets. Several innovative strategies have been developed to address this challenge. One such approach is "transfer learning", which utilizes pre-trained models—originally trained on large, labeled datasets—and fine-tunes them on smaller, less labeled datasets, thereby reducing the need for extensive labeled data in the new context. Another effective method is "data augmentation", which artificially expands the training dataset by creating modified versions of the available

data. This technique helps improve the model's ability to generalize without requiring additional labeled samples. Together, these strategies offer powerful alternatives to traditional methods, enabling more efficient training of DL models in scenarios where labeled data is scarce or difficult to obtain.

1.5.2 Training Deep Learning Models

DL models can be trained in both supervised and unsupervised ways. In supervised DL, labeled input data (e.g., an image of a diseased leaf) are mapped to an output (e.g., a specific plant disease) through a weights vector, with errors back-propagated from the output to adjust the weights. In contrast, unsupervised DL focuses on identifying patterns in the data for tasks like clustering or hashing. Training DL models typically involves Stochastic Gradient Descent (SGD) or variants like ADAM, using backpropagation to update model parameters. The choice of hyperparameters, including network architecture, learning rate, and activation functions, greatly influences model success. While architectures can be tailored to specific problems, it's common to start with a pre-established architecture that has proven successful in similar domains, such as AlexNet [94], VGGNet [182], and ResNet [78].

In addition to selecting appropriate hyperparameters, another crucial consideration during deep learning (DL) model training is the issue of computational costs. Training large and complex DL models requires significant computational resources, especially when dealing with extensive datasets and intricate model architectures. Factors such as model size, the volume of training data, and the number of epochs directly influence the computational load and training time. Large models with numerous layers and parameters necessitate more computations per training step, resulting in longer processing times. Additionally, specialized hardware like Graphics Processing Units (GPUs) or Tensor Processing Units (TPUs) is often required to accelerate training, as they are designed to handle parallel computations more efficiently than traditional CPUs. These increased computational demands also lead to higher energy consumption, which can be costly and raise sustainability concerns.

Optimizing computational efficiency is crucial, as it not only helps reduce expenses but also enhances the accessibility and scalability of model development. By improving computational efficiency, we can make advanced deep learning models more feasible to train and deploy, even in resource-constrained environments. This

is why modifying the architecture of models is essential—adapting architectures to balance performance with resource requirements allows for more efficient use of computational resources, facilitating the broader application of these models across diverse contexts and industries. Some of the most popular architectures in this domain include Region CNN (RCNN) [67], Fast RCNN [67], and You Only Look Once (YOLO) [176].

YOLO (You Only Look Once) is an efficient object detection model that, once trained on labeled data, can predict the presence of specific traits or features in new, unseen images. In applications like plant phenotyping, these traits could include signs of stress, diseases, or other physiological conditions. YOLO detects and classifies these traits in real-time, drawing bounding boxes around them. This ability to analyze unknown images, not part of the training dataset, makes YOLO a valuable tool for large-scale, automated monitoring and real-time predictions, particularly in fields like agriculture and plant health research. Despite its advantages, YOLO models face challenges, particularly with detecting small objects and handling localization errors, especially in dense or partially obscured regions. There is also a trade-off between speed and accuracy, with earlier versions prioritizing performance over precision. To address these issues, newer versions like YOLOv5 and YOLOv8 improve feature extraction and detection, enhancing accuracy for small objects. Solutions such as multi-scale networks, and anchor box refinement further boost performance. While YOLO excels in real-time detection, continuous improvements in architecture, dataset diversity, and advanced techniques are needed for better accuracy, especially in complex applications like plant phenotyping.

1.5.3 Specific objectives

Previous research has significantly contributed to unravelling the complexities of plant phenotype. However, notable gaps persist within the literature. This section endeavours to bridge these gaps by meticulously scrutinizing the current corpus of knowledge and revealing novel perspectives to propel our comprehension forward.

The research gaps and defined specific objectives of this thesis can be succinctly outlined as follows:

- The CNN models have been developed to the simultaneous detection of plant traits, such as nodes, flowers, and fruits, using YOLOv5. The research utilizes

RGB data with relatively low resolution, highlighting the innovative nature of the methodology, as it operates effectively with "limited resources." The approach is scalable, with the potential for low-cost hardware, offering a feasible solution for widespread adoption in plant phenotyping and stress monitoring.

- This study introduces a novel data-balancing approach to address the challenge of imbalanced class distributions among plant traits. By implementing this strategy, the model receives a more balanced representation of each class during training, leading to enhanced detection performance across all categories. Additionally, the YOLOv8 deep learning model is enhanced by incorporating Squeeze-and-Excitation (SE) blocks into its architecture. This modification enables the model to dynamically recalibrate feature responses, significantly improving its ability to detect flowers, fruits, and nodes in tomato plants.

- This study introduces an innovative approach to enhance the detection of small plant components, such as fruits and flowers, by modifying the architecture of the YOLOv8 model. The primary innovation lies in integrating a shallower layer (P2) into the model's head. This integration allows the model to focus on low-level features, which are crucial for accurately detecting small objects that may lack distinct features and are easily obscured by the background. By incorporating the P2 layer, the model can better capture these essential details, leading to improved detection performance for small plant components. Another is introduc an integrated SO-YOLOv5 model, combining elements from both the YOLOv5 backbone and the YOLOv8 head. This integration aims to optimize performance by leveraging the strengths of both architectures, resulting in a model that is both efficient and effective in detecting small plant components. By modifying the model architecture and integrating these components, the study achieves enhanced detection accuracy for small plant traits while minimizing computational costs and simplifying the model's dimensions.

- The innovation of this approach lies in the development of a lightweight network architecture specifically designed for Root Systems Architecture (RSA) segmentation in resource-constrained environments. Unlike existing solutions, which often require high-resolution cameras or specialized sensors, the proposed model can effectively predict "root pixels" in cluttered images

captured with minimal camera specifications. This flexibility, coupled with the model's ability to operate without imposing strict constraints on imaging conditions, makes it adaptable and efficient. The simplicity of the architecture reduces computational demands while maintaining performance, providing a novel solution to RSA segmentation in environments with limited resources.

1.6 Thesis Outline

The following section furnishes a meticulous delineation of each chapter, with the primary aim of achieving the research objectives delineated within this study. Elaborate expositions of the methodologies and procedures employed to fulfil these objectives are expounded upon within their corresponding chapters. The present thesis is organized into seven discernible chapters, as delineated below:

- Chapter 1. The introduction offers an extensive exploration of the background context and underlying motivations that propel the study forward. It is primary function is to acquaint the reader with the research problem, providing a rationale for it is importance within the academic discourse. Additionally, this chapter articulates the precise objectives of the research and presents a detailed overview of the thesis structure, outlining the organization and progression of subsequent chapters. Ultimately, Chapter 1 serves as a scholarly preamble, establishing the framework for the subsequent analytical investigation.

- Chapter 2. This chapter provides an in-depth exploration of the current landscape of plant phenotyping methodologies and the associated challenges. It explains the factors influencing high-throughput plant phenomics, discussing limitations related to hardware, such as various platforms and sensors, as well as constraints in traditional software approaches and algorithms. Additionally, it describes the transformative advances in technology that have led to the development of new, non-destructive, and versatile methods.The review also identifies key challenges in data acquisition, processing, and analysis, highlighting the need for efficient and accurate analytical methods. Furthermore, it identifies research gaps in the field of plant phenotyping, setting the stage for the research presented in the rest of the thesis. A systematic review investigating the impact of hardware and software variables on high-throughput

plant phenotyping, utilizing the findings presented in this chapter, has been published in [188].

- Chapter 3. In this Chapter, this thesis outlines the materials and methods employed, providing a detailed discussion of the research type and project design. It elucidates the nature of the data utilized, emphasizing the crucial steps of data preprocessing and preparation for thorough analysis. Additionally, it comprehensively covers the tools and software employed throughout the study. Finally, this chapter elaborates on the evaluation criteria applied to assess the efficacy of the models developed.

- Chapter 4. This chapter introduces an advanced framework for detecting plant traits using single-stage detectors, either independently or as part of an ensemble, based on YOLOv5. It focuses on addressing challenges posed by annotation-limited datasets, aiming to enhance the identification of nodes, fruits, and flowers within datasets acquired during stress experiments conducted across multiple tomato genotypes. Test-time augmentation (TTA) and model assembling are used to improve the proposed models. Furthermore, the models are subjected to testing with different backbones to analyze the impact of various network layers and configurations on the achieved results. The results presented in this chapter have been published in [24].

- Chapter 5. This chapter introduces an innovative approach to data balancing techniques to address dataset imbalances. By amalgamating data and generating supplementary samples for underrepresented classes, the dataset is rebalanced. Additionally, to enhance the recognition capability of targeted classes presents an attention module into the head architecture of the proposed model (YOLOv8). This chapter also presents a novel approach to dealing with the challenge of information loss while still taking advantage of data balancing. The results presented in this chapter have been published in [187].

- Chapter 6. This chapter presents a comprehensive analysis proposed to optimize model head adaptations aimed at enhancing the detection of small objects, building upon the base architecture of YOLOv8. Additionally, this chapter introduces an Integrated model from YOLO family models (Amalgamating components from both the YOLOv5 and the YOLOv8) to minimize computational costs and model dimensions.[186]

- Chapter 7. This chapter presents an alternative approach to the RSA segmentation problem that eliminates the need for a U-shaped grid, instead utilizing binary classification through probabilistic map estimation to classify the pixels of the original image as either background or foreground. Although U-shaped models are effective, they can be overly complex for this classification task and only provide a binary output, rather than assessing the probability of observing rooted or unrooted parts of an image. To address this issue, this work introduces a processing pipeline for the end-to-end analysis of plant RSAs. The results presented in this chapter have been published in [25].

- Chapter 8. Conclusions and future works for Future Research presents the study contributions and summarizes it is key findings. It also addresses the limitations encountered, providing context and insights into the scope and reliability of the research. Finally, the chapter identifies potential areas for future research, offering a concise overview of the study and it is implications for the field.

Chapter 2

Literature Review

2.1 Overview

In the preceding chapter, the advent of a novel frontier in phenotyping known as high-throughput phenotyping (HTP) was delved into. This emerging discipline harnesses modern data sampling methods to amass vast datasets, poised to revolutionize the efficacy of phenotyping. Numerous research collectives acknowledge the immense potential of HTPP and have committed substantial resources to its infrastructure or are contemplating such investments. To maximize the utilization of scarce resources, meticulous planning and judicious utilization of these facilities are imperative, coupled with carefully interpreting the results.

The literature review chapter is planned with the aim of providing an overview of the current state of research on Hardware and software factors affecting High-Throughput Plant Phenotyping. In examining each of these domains, our scrutiny reveals two pivotal determinants. In the hardware domain, our study encompasses plant phenotyping platforms and sensing apparatus. In contrast, in the software domain, our attention will be drawn towards algorithms for analyzing data obtained from hardware tools. These factors were selected as they refer to two main categories: hardware (platforms and sensors) and software (algorithms and new trends), as reported in Figure 2.1.

Hardware tools are designed to minimize costs associated with plant phenotyping and facilitate non-destructive characterization of desired traits, thereby advancing the field of high-throughput phenotyping (HTP). HTP platforms automate trait

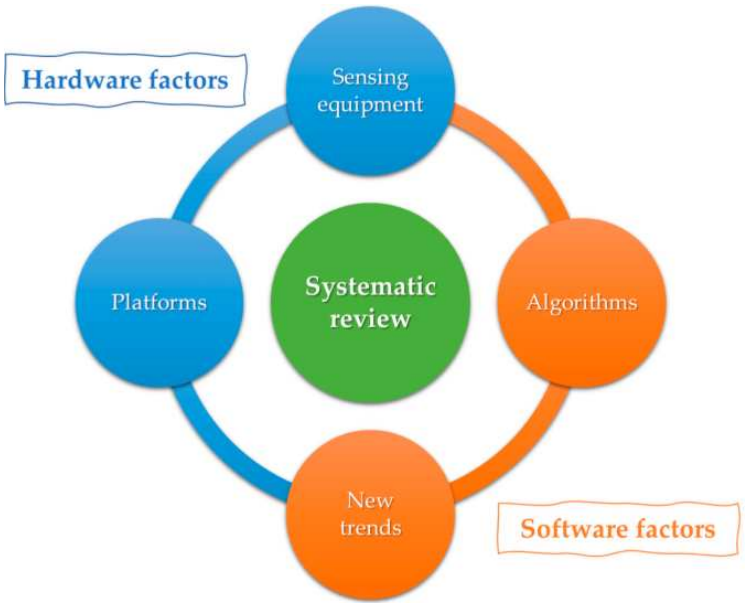


Fig. 2.1 Diagram of the two main categories—hardware and software—considered in this review, as well as the four factors specifically addressed: platforms, sensing equipment, algorithms, and new trends. Blue: hardware-related factors; orange: software-related factors.

characterization, saving time and effort while safeguarding plants from damage. The integration of advanced technologies and novel sensors with these platforms is pivotal in enhancing data collection processes.

Seamlessly merging insights from phenotyping with the realms of computer science, engineering, and data analysis, HTP epitomizes a convergence of disciplines. Central to this synergy are machine learning (ML) and deep learning (DL) algorithms, seamlessly integrated with non-invasive imaging modalities. These algorithms stand as linchpins in the automation, standardization, and analysis of quantitative data, heralding a new era of precision in phenotypic exploration.

Since the factors described above are highly interdependent, this section explores these interconnections, emphasizing the relationships between hardware and software, such as platforms, algorithms, sensors, and emerging trends in data processing. A systematic study, guided by the PRISMA protocol, was conducted using a literature search strategy with defined inclusion and exclusion criteria. This was followed by a quality assessment, leading to the extraction of relevant data [139]. Hence, the first step was to gather studies on HTP by performing an extensive literature search. As such, the Scopus database was used for two main reasons: According to [169],

using more than one database for a literature search does not guarantee a positive impact on the research outcome.

The high degree of reliability of Scopus guarantees the evaluation of high-quality papers published in qualified journals. The research considered only papers published between 2019 and 2022, aiming to include only recent relevant publications. The following keywords were used to search for articles using AND/OR operators: “Sensor” AND “high-throughput plant phenotyping”. “Machine learning” OR “deep learning” AND “high-throughput plant phenotyping”. “Platform” AND “high-throughput plant phenotyping”. “Image acquisition technique” AND “high-throughput plant phenotyping”. About 1000 published and in-progress articles were found at the first stage before applying filter restrictions (based on language, abstracts, articles not published in full, and articles not related to the investigated topic). Then, about 500 papers out of 1000 were filtered. Figure 2.2 shows a PRISMA flowchart detailing the extraction of articles relevant to the study and the subsequent filtering stages that were applied.

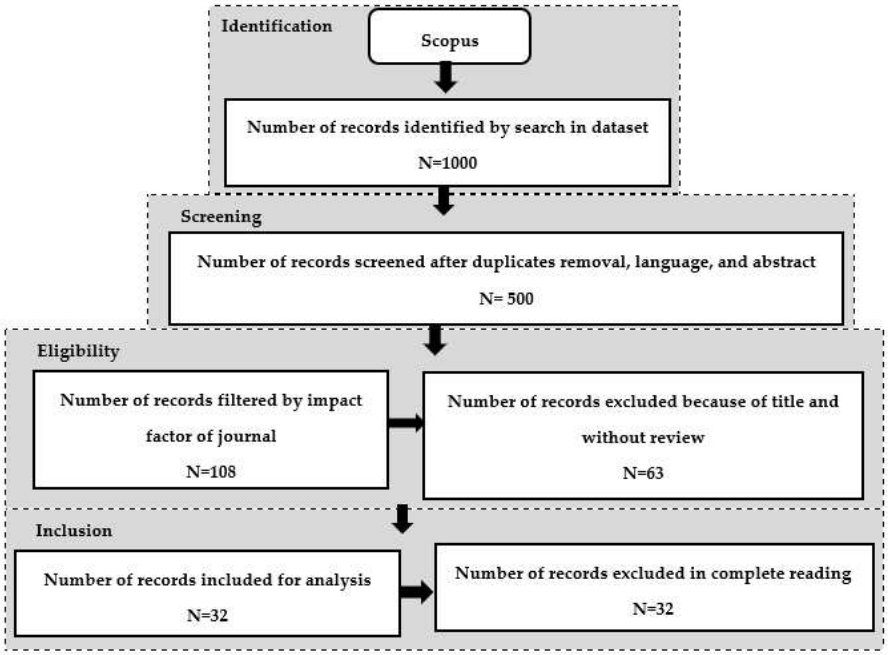


Fig. 2.2 Flow diagram of database search using PRISMA.

The papers were selected through a combined inclusion/exclusion test conducted according to the PRISMA protocol. Specifically, Table 2.1 highlights the exclusion criteria, while Table 2.2 describes the inclusion criteria. These criteria were used to

filter the 500 papers acquired after the screening step, resulting in a final result of 32 relevant items representing the final database used for the review analysis.

Table 2.1 Exclusion criteria

#	**Exclusion Criterion**
1	Articles not written in English
2	Articles that do not refer to high-throughput plant phenotyping
3	Articles that relate to phenotyping traits but are unrelated to the discussion
4	Articles that do not use DL or ML
5	Articles that appear in invalid journals or those with very low-impact factors
6	Articles that are reviews
7	Articles for which only abstracts are available

Table 2.2 Inclusion criteria

#	**Inclusion criteria**
1	Articles written in English
2	Articles that refer to the high-throughput plant phenotyping
3	Articles that use DL or ML
4	Articles that appear with high-impact factors
5	Research articles (non-review papers)
6	Articles that are fully available
7	Articles that relate to selected research questions

The collected and filtered items were carefully ranked according to their merits and categorized with respect to the research questions identified beforehand. The data extraction procedure is summarized in Table 2.3.

Table 2.3 Data extraction

#	Element Contents	Type
1	Title	Yes/no
2	Research questions	The clear description of the research question
3	Type of article	Problem identification
4	Study outcomes	Short description of study outcomes
5	Year	The year of publication
6	Journal	Journal Impact factor (Q1)

In this thesis, the focus will be on the software domain, where the objective is to delve deeper into the intricate landscape of algorithms utilized for analyzing data obtained from hardware tools in the realm of high-throughput phenotyping (HTP). By scrutinizing the latest advancements and trends in software development tailored for HTP applications, the aim is to unravel the nuanced interactions between algorithms, data processing methodologies, and their impact on enhancing the efficiency and accuracy of phenotypic analysis. Through this exploration, a contribution is made to the burgeoning body of knowledge surrounding software factors influencing HTP, ultimately fostering advancements in the field and paving the way for future innovations in phenotypic research.

2.2 Hardware factors in High-Throughput Plant Phenomics

2.2.1 High-Throughput Plant Phenomics Platforms

In recent years, a wave of specialized hardware tools has emerged, poised to enhance the efficiency of plant phenotyping processes while concurrently mitigating time constraints and associated costs [224]. Modern fully automated systems can now efficiently screen hundreds of genotypes and thousands of individual plants or field plots using non-destructive sensors. The data collected is automatically processed and stored for future analysis [192], [119]. Among these tools, an array of phenotyping platforms has arisen since the early 2000s, now firmly established as indispensable

apparatuses within both commercial and academic research domains [70]. These technological advancements have facilitated non-invasive trait analysis, paving the way for the evolution of a burgeoning field known as high-throughput phenotyping (HTP).

A hallmark of a high-throughput phenotyping platform (HTP) lies in its capacity to capture imagery of hundreds of plants daily [51]. While some HTPs rely solely on imaging techniques, others integrate non-destructive contact-based methods. In this context, "HTP" denotes a platform engineered to efficiently gather extensive phenotypic data from hundreds of plants daily, employing a high level of automation. This system serves as a robust tool facilitating rapid, non-destructive, and high-throughput monitoring and quantification of crop growth and production-related phenotypic traits. Consequently, by streamlining processes and curtailing time and labor requirements, HTP minimizes the potential for destructive harm to plants.

In the forthcoming years, an increasing number of such systems will be developed to efficiently screen a vast array of plants for attributes such as size, growth patterns, and various other traits. Nonetheless, akin to any intricate machinery, the utilization of these platforms poses challenges and limitations that may be underestimated by those unfamiliar with them. Hence, before committing resources to acquiring, developing, and utilising such platforms, it is imperative to meticulously assess potential issues and strategize on how to overcome them. For example, the prevalent development of large-scale phenotyping platforms by professional commercial entities has led to the safeguarding of underlying hardware and software components through patents, thereby limiting modifications to meet specific research requirements [40]. This becomes particularly crucial when multiple research groups with varied interests are involved, as there is a risk that the platform may end up as a compromised solution that fails to meet anyone expectations. Ensuring effective communication and alignment of objectives is essential to mitigate this risk and guarantee the success of the project.

Each platform incorporates a diverse array of sensing equipment, including unmanned vehicles, drones, and satellites [53]. The selection of sensors is intricately tied to the application context in which the particular high-throughput phenotyping platform is intended to operate. For example, as depicted in Figure 2.3, aerial platforms typically offer superior flexibility and efficiency compared to ground-

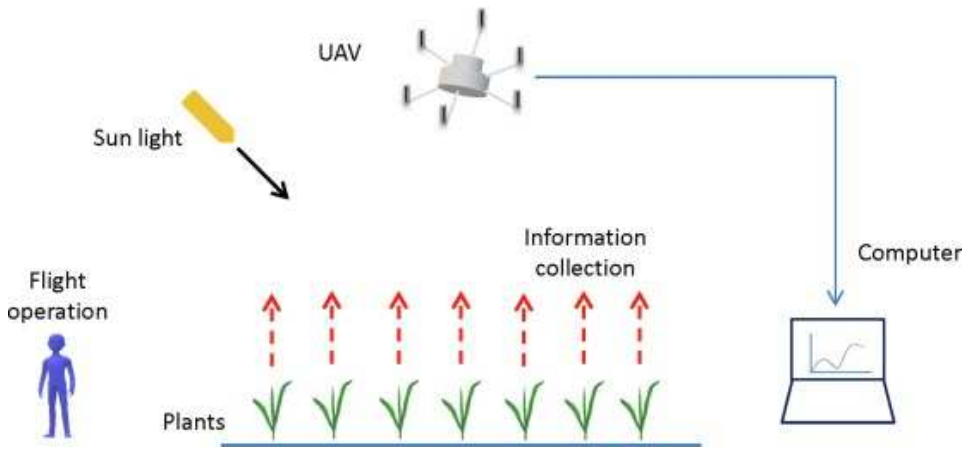


Fig. 2.3 Aerial platform data collection for plant high-throughput phenotyping in open field [216].

based platforms, making them particularly suitable for deployment in open areas [84].

Conversely, ground-based robotic platforms may excel in the longitudinal monitoring of plant traits within enclosed environments [8]. Alternatively, satellite-based platforms can prove highly effective when extensive data collection is required over vast geographical regions [232].

Within the realm of platforms, research has underscored the effectiveness of aerial platforms, particularly unmanned aerial vehicles (UAVs), in furnishing real-time, precise insights into the physiological parameters and geometric characteristics of plants across expansive areas, while accommodating a diverse range of sensors [55],[149]. The versatility of these platforms lends itself to a myriad of applications, offering users straightforward operation without the need for extensive training [55]. Nonetheless, UAV imagery may be susceptible to environmental variables such as temperature, wind, humidity, and precipitation, which can compromise data quality [55]. Additionally, UAVs face challenges related to battery life limitations, necessitating meticulous route optimization to maximize data-gathering efficiency.

Another crucial consideration, as highlighted by the authors in [217], is the potential for cost reduction in phenotyping efforts through the implementation of automated platforms, especially if these robots are engineered to be versatile and adaptable to various crop types. There is a need to collect recurring measurements of plant characteristics across extensive populations at multiple intervals throughout

a growth season. In such scenarios, robotic systems emerge as highly desirable tools due to their ability to provide the necessary speed and precision required for phenotyping endeavors of this nature. This involves accommodating factors such as the diverse appearances, morphologies, growth stages, and sizes of different plant species. Robotic platforms for plant phenotyping can be categorized into indoor and outdoor systems. In controlled environments, plants are typically stationary while robots move around them or are transported to fixed locations. [121] utilized a robotic arm with a TOF camera to measure the stem height and leaf length of corn seedlings. [28] developed a gantry robot with a 3D laser scanner to analyze the surface area and volume of Arabidopsis and barley plants. Both encountered challenges in capturing obscured plant parts, a common issue in image-based phenotyping. To address this, [214] proposed an automated multi-robot system with depth cameras, employing deep learning for optimal viewpoint selection to overcome occlusion more efficiently. However, accurate predictions from deep networks are crucial for determining these viewpoints.

Controlled environments offer plant growth and trait quantification advantages, but plants in such conditions often differ from those in field settings due to environmental influences. Thus, field-based phenotyping yields more actionable insights for crop improvement except for certain horticultural crops. Various platforms have emerged for field-based high-throughput phenotyping, navigating between crop rows to approach plants. This challenges navigation and data collection, including factors like temperature and soil unevenness. Hence, robotic systems for field phenotyping must withstand these challenges, featuring mobile platforms equipped with all necessary components for navigation and data acquisition. These platforms rely on GPS data and/or sensors to navigate effectively through the field environment, ensuring efficient phenotypic assessment. Unmanned Ground Vehicle (UGV) robotic systems utilize various sensors including LIDAR and cameras (RGB, TOF, NIR, stereo vision) for data collection, either fixed on a stand within the mobile platform or attached to a robotic arm for increased versatility. Techniques such as 3D reconstruction, image processing, and machine learning analyze data to quantify morphological traits. These systems measure diverse traits across multiple plant species and agricultural contexts, including plant height, leaf area, and canopy architecture. For instance, the TerraSentia [60] rover captures maize plant scans to derive Latent Space Phenotypes (LSPs) using machine learning from RGB and LIDAR data. Vinobot [180], equipped with a six DOF robotic arm and 3D imaging sensor, measures maize and

sorghum height and leaf area index (LAI). However, semi-autonomous navigation poses challenges, requiring alignment with crop rows before data collection. Figure 2.4 illustrates robotic platforms in an open environment.

Fig. 2.4 Ground-based robotic platforms for plant high-throughput phenotyping in open field [8]; A: Vinobot [180], B: Robotanist[145], C: A robotic system to slide LeafSpec across entire leaf to collect its hyperspectral images[34], D: Thorvald II [72], E: BoniRob [15], F: Ladybird [197], and G: Flex-Ro [146].

Figure 2.5 shows the distribution of the studies in 32 summary studies from 2019 to 2022, according to the part of the plant under analysis. Specifically, 24 studies (i.e., 75% of the total number of papers) focused on the aerial parts of plants, including leaves, flowers, and fruit. Meanwhile, 19% of reviewed papers concerned root systems, and only two studies (i.e., 6% of the total papers) were conducted on seeds.

To better understand the strengths and weaknesses of different high-throughput plant phenotyping platforms, the papers were grouped by platform type, as shown in Figure 2.6. Five different platform types were categorized: ground platforms, aerial platforms, root platforms, vehicles, and microscopic platforms, showing that almost one paper out of two focused on ground platforms.

The plant characteristics to be evaluated suggest using a specific HTP platform instead of another. For example, some platforms are specifically tailored to acquire

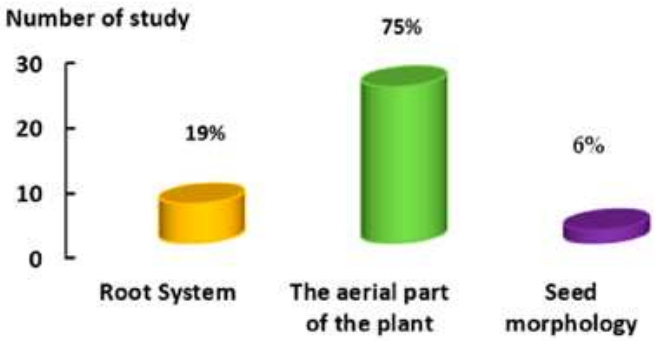


Fig. 2.5 The number of studies (on the Y-axis) according to the specific part of the plant under analysis (on the X-axis). As can be seen, 19% of the studies focused on the root system architecture, in contrast to 75% on the aerial part of the plant, and 6% on the morphology of the seed

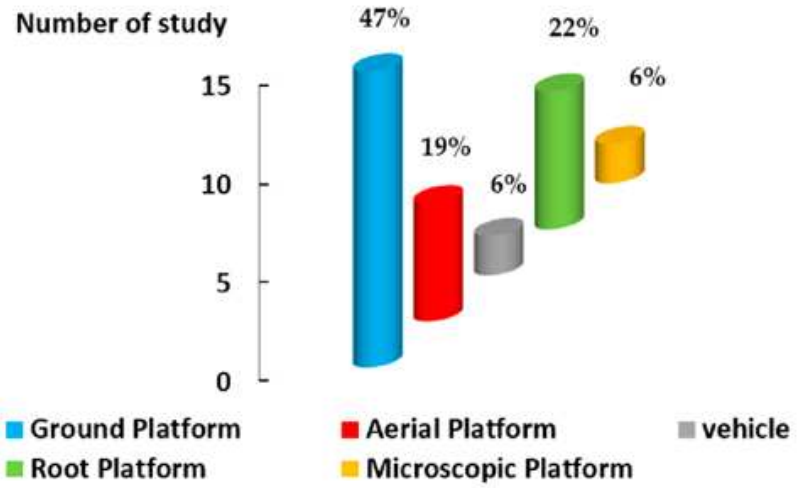


Fig. 2.6 The number of studies (on the Y-axis) according to the platform used for HTP (on the X-axis). The distribution shows that most of the studies (about 47%) used ground platforms, followed by root platforms (used by 22% of the studies) and aerial platforms (about 19%). Only 6% of the reviewed papers used vehicles and microscopic platforms.

images concerning the aerial parts of the plant, while others can target the roots; furthermore, some can be used in laboratories and greenhouses under controlled conditions, while others are specifically designed to be deployed directly on the field. Ground platforms were the most used due to their flexibility, ease of use, and relatively low cost. Among the studies analyzed, it is worth mentioning PhenoTrack3D [44], a pipeline able to reconstruct the 3D architectural development of a maize plant at the organ level throughout its entire lifecycle, as well as the method described in [48], where authors proposed an HTP platform based on CNNs to detect pots and segment lettuces in the greenhouse environment automatically.

Although ground platforms are flexible and easy to use, they may not be well suited for operating in open fields. Consequently, some researchers shifted their focus toward aerial platforms, which encompassed 19% of the reviewed studies. Intuitively, unmanned aerial systems could provide an improved assessment of plant phenotype when used in large-scale environments. For example, the authors in [156] showed that aerial platforms have more control over the crops than ground platforms, providing data captured from points of view not reachable by ground robots. Aerial platforms can also incorporate spectral sensors, exploiting the information on spectral wavelengths to improve the identification of plant organs [236]. Despite the advantages, however, aerial platforms must deal with the constraint limited by adverse weather conditions [92],[156]. The introduction of vehicles and robots to perform HTP activities has greatly accelerated the whole process. Among the reviewed papers, two studies focused on agricultural vehicles, achieving promising results [87],[133]. Specifically, in [87], vehicles that simultaneously capture a plant from four different angles were used, allowing a complete characterization of a single plant per run. The discussion needs to be moved to the platforms aimed at studying the root system architecture (RSA) of a plant, covered by about 22% of the reviewed papers. The RSA contains the growth information that allows revealing the health status of the plant [68]. As such, phenotyping on root systems provides researchers with a useful tool to identify which plants can perform better [194]. However, as roots are usually covered by soil, which does not allow a direct visual assessment, different tools are required to extract visual information from these systems (e.g., X-ray cameras) and, consequently, to process these data. This highlights both the challenges posed by this topic and the focus of the scientific community on overcoming these issues due to the importance of developing proper automated RSA monitoring and assessment tools. Some examples are given below. The authors in

[123] proposed a low-cost hardware system with a controlled chamber to inspect root characteristics, automatically performing image preprocessing, feature extraction, and segmentation on gathered data. The authors of [123] proposed a mobile tool that automatically monitors root growth in a laboratory environment. Let us point out that neither of these platforms handles the challenges related to acquisition settings, such as illumination and occlusions. As such, the authors in [235] proposed a fully automated, customizable, embedded platform that deals with each stage of the root development cycle. This platform allows a rapid assessment of root morphology and growth rate, improving the overall effectiveness of root HTP. To answer RQ1 about root platforms, it must be noted that the platforms analyzed for RSA are suitable only in laboratory environments. Lastly, 6% of the reviewed papers studied plant leaves using a microscopic platform. Phenotyping of leaves, including assessment of their morphological characteristics, allows a better understanding of their operation and function [66]. Microscopic platforms are a good case for studying leaf morphology. They can image the areas of leaves on both sides with their cameras to automatically determine the stomatal index [239] or phenotype hairy leaves; a microscopic platform can be a simple, powerful, and inexpensive imaging method [170]. However, when phenotyping leaves, attention must be paid to the magnification settings of these platforms, as this measure may positively affect the detection of specific leaf characteristics.

In the forthcoming chapter, a comprehensive explanation of the chosen platform in this thesis will be provided.[3.1]

2.2.2 Image Acquisition Sensors

Remote sensing is a sophisticated technique used to understand and monitor the physical properties of a specific area or object by analyzing the radiation it reflects and emits from a distance. This methodology employs a diverse range of sensors to collect data swiftly and without intrusion. These sensors are adept at capturing various segments of the electromagnetic spectrum, as depicted in Figure 2.7. They meticulously measure the flux of solar radiation across a spectrum ranging from short to exceedingly long wavelengths, rebounding from the points of interest. The reflectance of vegetation is closely tied to its physiological, biochemical, and structural characteristics, particularly focusing on leaf composition. This reflectance is influenced by three primary physical processes: absorption, reflection, and transmis-

sion. Changes in leaf structure, whether caused by factors such as plant genotype, leaf age (e.g., senescence), or variations in health and physiological conditions (e.g., light intensity, nutrient levels, or water availability), can significantly affect the absorption, transmission, and reflection of electromagnetic waves. These changes result in fluctuations in the spectral signature [138], [73].

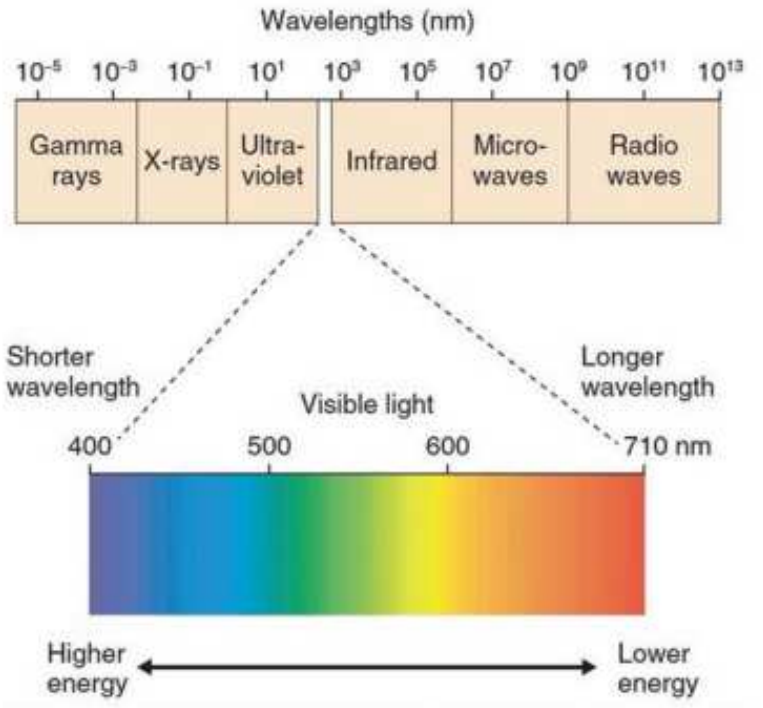


Fig. 2.7 The electromagnetic spectrum includes the optical spectrum, featuring the visible and ultraviolet regions [2].

Imaging sensors exhibit great potential for early stress detection by swiftly, non-invasively, and efficiently capturing plant images, thereby documenting nuanced shifts in phenotypic traits [63]. These sensors are ubiquitous, seamlessly integrated into a myriad of electronic devices like smartphones and digital cameras, facilitating the capture of images and videos. The evolution of imaging sensors has been marked by a steady progression over the years. When employing a camera system, one must decide on the number of wavebands to incorporate, ranging from three to dozens, and determine the minimum required spatial resolution.

While RGB sensors, which cover the visual range of 400-700 nm with three broad bands: red, green, and blue, may offer less spectral information compared to multispectral and hyperspectral sensors, they compensate by providing data with

exceptional spatial resolution [18]. Renowned for their high resolution and color depth, RGB sensors find extensive utility in remote sensing applications, particularly in monitoring variations in plant leaf color or texture [69]. Their popularity among researchers stems from their capability for detailed visual analysis of growth, plant colouration, and morphometry. The cost-effectiveness and adaptability of RGB sensors in both controlled settings (such as glasshouses and growth chambers) and field environments render them a preferred choice where budget and versatility are paramount concerns. Furthermore, their adaptability across diverse crop types and environmental conditions cements their status as the most versatile tool in agricultural settings. Utilizing a range of wavelengths for imaging, both RGB and spectral sensors excel in pinpointing subtle changes within plant tissues, imperceptible to human vision [201]. They have diverse applications, from identifying grapes [133] to enumerating plant traits [120]. The Multiple RGB sensor systems enhance precision, though they may overlook neighbouring plants. To address this, an aerial acquisition setting is proposed, which with RGB imaging extends applications in plant localization [92]. Nonetheless, RGB sensors pose distinct challenges. While their accessibility and simplicity are advantageous, their restricted spectral data presents a hurdle for a thorough analysis of physiological complexities induced by plant stressors [106].

To address the limitations inherent in RGB sensors, spectral imaging sensors have emerged as a favored option among researchers. These sensors gather and analyze data across numerous narrow spectral bands, offering enhanced insights into plant physiology [179]. Unlike RGB sensors, spectral imaging delves into the plant spectral signature, facilitating the identification of chemical and biological properties that remain undetectable otherwise. They also enable a more thorough assessment of plant traits compared to RGB sensors [238]. Operating beyond the visible spectrum (300 to 900 nm), spectral sensors capture images using wavelengths with high spectral resolution for each pixel, providing a comprehensive overview of the plant electromagnetic spectrum. Through the analysis of the plant spectral profile, spectral sensors can discern changes in key health indicators such as chlorophyll concentration or water content, imperceptible to RGB sensors. This capability holds promise in identifying stress indicators in their nascent stages and tracking their progression over time[228]. In a study [236], they showed improved accuracy in detecting plant organs. [11] contend that drones equipped with multispectral sensors demonstrate superior capability in detecting radiation changes under low-light conditions, and

autonomously determining optimal flight altitudes for crop estimation. While the adoption of spectral imaging often necessitates deeper analysis despite its higher costs and data processing demands, it proves particularly advantageous for scrutinizing specific crops exhibiting subtle physiological changes under stress.Nonetheless, despite spectral imaging advantages, this imaging entails higher expenses, especially for sensors with superior spectral resolution (900 to 2,500 nm) [201]. Furthermore, the technique generates a substantial volume of data necessitating specialized software for processing and analysis. Even with such software, the procedure can be labor-intensive, requiring expertise in data analysis [127]. Despite advancements in spectral imaging, fluorescence imaging is favored for its ability to detect subtle changes in plants' attributes, aiding in stress identification. However, the need for dark adaptation adds complexity to setups [71], also fluorescence imaging is less common due to cost and specificity limitations, making RGB-based methods more prevalent.

Thermal imaging sensors, unlike traditional RGB sensors, detect infrared radiation emitted by objects, providing valuable insights into plant health by monitoring temperature fluctuations [38]. Particularly in agriculture, they aid in detecting stress factors like water scarcity or disease, enhancing crop management practices [201]. Despite its benefits, thermal imaging is less common in plant stress research compared to RGB or spectral sensors, primarily due to the complexity and expense of thermal imaging equipment. High-quality thermal sensors are often costly, hindering accessibility. Additionally, interpreting thermal imaging data accurately demands specific expertise, as environmental factors like humidity and wind can influence temperature readings[65].

Moreover, the utilization of various sensors such as multi-spectroscopy, Raman spectroscopy, magnetic resonance imaging (MRI), and X-ray imaging has been pivotal in scrutinizing the internal anatomy of fruits and seeds, as well as the overall internal morphology of plants [81, 104, 41, 91, 198]. Particularly, X-ray imaging has revolutionized the study of root development, surmounting challenges linked with visually evaluating root system architectures amidst soil interference [57, 80, 9]. Furthermore, specialized imaging systems like rhizotubes or minirhizotrons buried underground have been investigated for capturing high-resolution root images, incorporating guided scanners and camera systems [13].

The papers in the review pool, spanning 32 summary studies from 2019 to 2022, utilized four main types of sensors. The choice of sensors largely depended on the research problem and the desired quality of the collected data. Thus, there is a need for a better understanding of the kinds of data (and subsequent plant traits) usually captured and studied by the domain experts. The proposed categorization includes RGB, multispectral, and hyperspectral cameras, and X-ray CT. Figure 2.8 shows the distribution of sensor types within the reviewed studies.

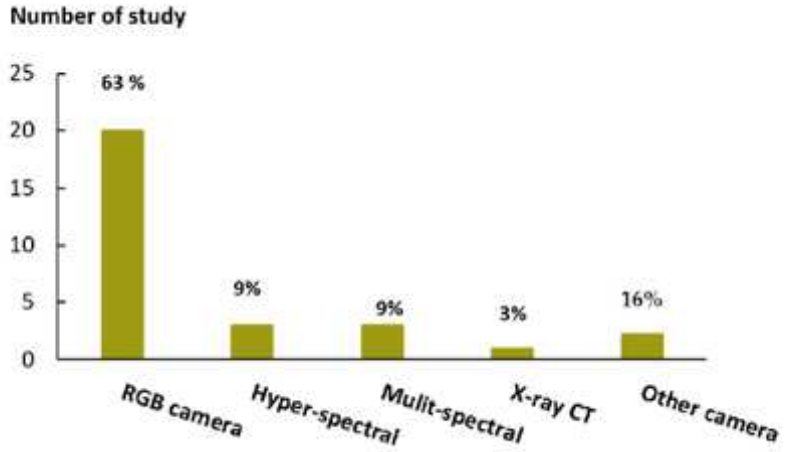


Fig. 2.8 The number of studies (on the Y-axis) according to the sensor equipment used for HTP (on the X-axis). The distribution shows that most of the studies (about 63%) used RGB cameras, followed by hyperspectral and multispectral cameras (both at 9%). Few researchers used X-ray CT, while approximately 16% of papers used other types of sensors

About 63% of reviewed studies used RGB sensors to capture plant images. These sensors are the most popular for capturing the morphological information of plants due to their inexpensiveness and ease of use. RGB sensors operate within the range of human vision, with wavelengths ranging from 400 to 700 nm. For example, low-cost RGB sensors have been used to detect grapes and estimate their volume [133] or to count the aerial traits of plants, including ears and grains of cereals [120]. Systems composed of multiple RGB sensors have also been developed. For example, the authors in [87] used an array of four RGB sensors to improve the accuracy of a system used to count flowers. Let us note that this configuration poses unique challenges; for example, the authors highlighted that the sensor array can overlook images of neighboring plants. To overcome this issue, the authors proposed an aerial acquisition setting, where the array is used to look at a set of plants instead of a single one. Aerial gathering vehicles equipped with RGB images were also used in [92] to

gather data for plant localization. Multispectral sensors can also be used to gather images to evaluate the characteristics of plants due to the use of multiple spectral bands, which can provide more information if compared to RGB sensors [238]. The effectiveness of multispectral sensors was demonstrated in [236], where the authors showed improved accuracy in detecting plant organs. Drones can also be equipped with multispectral sensors, achieving improved results in detecting dynamic changes in radiation in low-light environmental conditions and automatically determining the optimal flight altitude for crop estimation [11]. Another type of sensor emerging in HTP is the hyperspectral sensor, which is able to capture images with a few nanometers of wavelength resolution, ranging from ultraviolet radiation to infrared. As for the reviewed papers, only three used hyperspectral sensors as it is quite an expensive sensing technology that emerged on the market in recent years. However, since a hyperspectral image is a collection of many images (depending on the spectral resolution of the sensor), hyperspectral data have huge dimensions and are complex to manage and process. The three papers showed promising results, especially when the data were analyzed using deep neural networks [50],[225],[165]. X-ray sensors have also been used for HTP to assess internal fruit morphology [198]. Therefore, it is fair to say that the only sensors capable of assessing the internal morphology of fruits and seeds to date are X-ray sensors performing CT scans. Lastly, microscopic sensors are suitable for visualizing microscopic leaf characteristics, such as stomatal leaf index and leaf fluffiness, achieving good accuracy [170],[239].

In recent years, the convergence of data from diverse imaging sensors via multimodal approaches has emerged as a promising strategy for comprehensive plant stress analysis [198]. Multimodal techniques amalgamate RGB, spectral, and other imaging sensors to capture a holistic image representation of a plant health condition. Although offering richer analysis potential, multimodal approaches pose challenges due to complexities in data integration and demands for sophisticated algorithms capable of effectively analyzing bundled datasets, thus potentially escalating computational requirements. Therefore, while promising enhanced analysis capabilities, multimodal approaches require careful consideration regarding sensor format compatibility and synchronization.

These challenges often lead researchers to opt for RGB sensors due to their cost-effectiveness, simpler operational requirements, and compatibility with widely available devices like smartphones and digital cameras.

2.3 Software factors in High-Throughput Plant Phenomics (Algorithms)

Recent advances in sensing technology have transformed data collection from High Throughput (HTP) platforms in the scientific community, shifting the focus from equipment selection to refining algorithms for data processing and image comprehension. Managing the voluminous data generated is crucial, but manual labeling and management are impractical due to their scale[68]. Consequently, recent methodologies leverage image-processing techniques to automate these tasks as much as possible[29]. Furthermore, the widespread adoption of highly precise methodologies, driven by the development of Graphics Processing Units (GPUs), has become feasible. These approaches, utilizing Machine Learning (ML), Deep Learning (DL), or a combination of both, achieve remarkable accuracy within reasonable timeframes and cost constraints[6].

The integration of computer vision algorithms with Machine Learning (ML), such as Support Vector Machines (SVMs), improved leaf classification accuracy [212], while a combined approach of computer vision, ML, and robotics achieved high accuracy (86%) in image segmentation tasks [20]. However, ML approaches struggle with low-quality images and fail to generalize in varied imaging conditions [86]. Deep Learning (DL) has garnered interest for it is ability to automatically learn features and generalize effectively, although it requires ample training data for tasks like segmentation and recognition [92],[1]. Researchers are comparing ML and DL methods to determine their suitability for evaluating plant traits, where ML excels in regression evaluations and DL may offer faster and more accurate classification due to it is superior feature identification [55]. In [135], authors proposed a method using SVM and a probabilistic active contour model for automatic plant segmentation and leaf counting, achieving 70% accuracy and reducing time efforts by at least 90%. Another study in [96] applied a Gaussian mixture model (GMM) for automatic root tip detection, reaching 97% precision in predicting the primary root.

While traditional ML methods have proven successful in phenotyping plants [135],[96], recent research suggests that Deep Learning (DL) can further improve outcomes [6]. In [3], authors compared various ML approaches (e.g., linear discriminant analysis, quadratic discriminant analysis, and k-nearest neighbours) with deep neural networks (e.g., AlexNet, VGG-19, ResNet-50, and ResNet-101) for

classifying watermelon seeds. The top-performing DL model (ResNet-50) achieved 87.3% accuracy, outperforming the leading ML model (linear discriminant analysis) which achieved 83.6%.

Current applications strive to alleviate the computational workload of manual labelers by implementing automated or semi-automated pipelines for estimating plant counts from imagery. This includes tasks like isolating target plants from noisy backgrounds in field images. For example, the KAT4IA pipeline introduced in[74] uses ML models and neural networks to extract plant pixels and estimate plant height. Similarly, in[52], a pipeline based on DL architectures like RetinaNet and UNet automatically identifies root locations in images. These automatic pipelines have been proposed for all data analysis steps, leading to decreased processing time and improved accuracy compared to manual methods [13],[44],[231],[239],[48]. Finally, the outcomes stemming from the fusion of Machine Learning (ML) and Deep Learning (DL) techniques prompted a comprehensive evaluation of these methodologies. A notable instance can be found in the study by [218], where researchers compared various ML and DL algorithms for image-based RSA phenotyping. These included k-means, naïve Bayes (NB), random forest, shallow neural networks, and deep neural networks. Results showed that the deep neural networks exhibited superior performance, achieving an accuracy of approximately 86% on a dataset consisting of 617 root images extracted from mature alfalfa plants.

In the domain of DL, substantial progress has been achieved using deep neural networks for object detection in HTP. Established architectures such as AlexNet, VGG16 and 19, and Inception have proven effective in detecting fruit in low-quality images [133]. Furthermore, innovative architectures have been investigated to tackle challenges in assessing and identifying plant phenotypic traits. For instance, a customized U-net model introduced in [123] delivered precise segmentation outcomes in seed recognition and plant root evaluation after being trained on a limited number of images.

Figure 2.9 shows the predominant approaches in 32 summary studies from 2019 to 2022. The discussion about data analysis and the use of algorithms is currently attracting the most attention. According to the articles selected for this review, 72% discussed data analytics and algorithms. This could be because, with the increasing amount of image data, there is an increasing need to develop powerful analysis tools capable of accurately and quickly assessing phenotypic traits. Thus, scientists are

trying to find ways to experiment with and analyze big data by integrating computer vision, ML, DL, and artificial intelligence approaches in general.

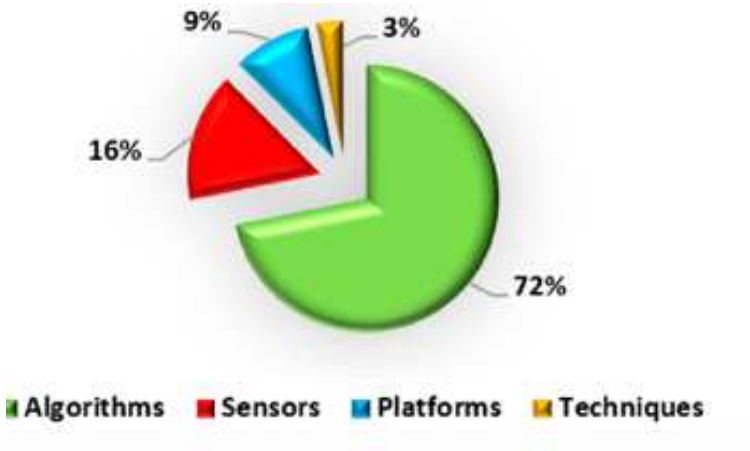


Fig. 2.9 Predominant approaches in 32 summary studies from 2019 to 2022

Lastly, it is worth analyzing whether specific families of algorithms are used by researchers to evaluate specific phenotypic traits. Figure 2.10 shows the distribution of studies according to the type of algorithm used by the paper authors. It can be seen that most researchers used DL to perform plant phenotyping activities.

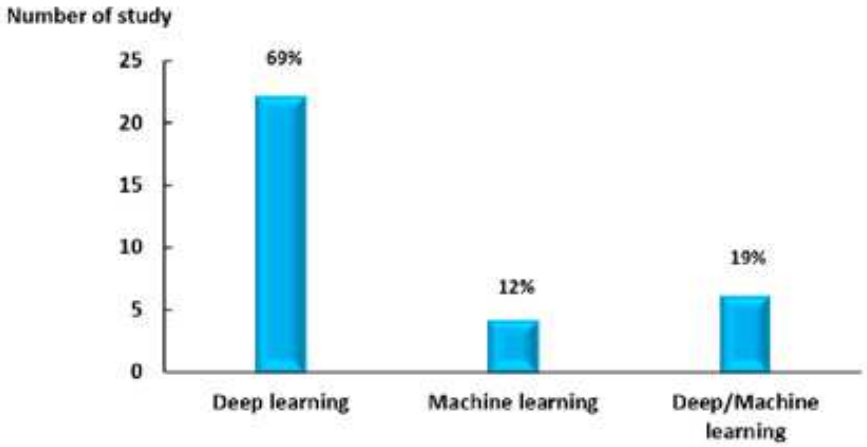


Fig. 2.10 The number of studies (on the Y-axis) according to the algorithm used for data evaluation (on the X-axis). Most studies (about 69%) used DL approaches, while only 12% of the studies were based on traditional ML. Lastly, 19% of the reviewed studies used hybrid approaches involving both ML and DL

ML and DL algorithms have been effectively used to analyze data from HTP platforms. ML algorithms have been used in several applications, such as plant seg-

mentation by K-means clustering [238], crop estimation via regression analysis using the Gaussian process and random forest [236],[162],[11], and the development of processing pipelines for the extraction and analysis of plant characteristics [236],[74]. Specifically, the latest application was developed to ease the computational burden on manual labelers, introducing automated or semi-automated pipelines to estimate the number of plants from existing imagery, e.g., separating target plants from a noisy background in field images. As an example, a self-monitoring pipeline called KAT4IA was proposed in [74] to extract pixels belonging to plants and estimate plant height by combining ML models and neural networks. Another approach for measuring root nodes was proposed in [52], where the authors created a pipeline based on DL architectures, such as RetinaNet and UNet [89], to automatically identify the location of roots on images. Let us underline that the idea of automatic pipelines has been proposed in all data analysis steps. Generally speaking, pipelines automating the analysis of the plant were able to reduce processing time, while delivering higher accuracy when compared with manual methods [13],[44],[231],[239],[48]. As for DL, an interesting development has been achieved via deep neural networks aimed at object detection applied to HTP. For example, existing architectures, such as AlexNet, VGG16 and 19, and Inception, were used to accurately detect fruit on low-quality images [133]. However, new architectures were also explored to deal with the specific challenges related to evaluating and identifying plant phenotypic traits. For example, the authors in [123] proposed a modification of the traditional U-net model to train a few images, achieving precise segmentation results in seed recognition and plant root evaluation. Object detection has also been performed using two-stages detectors, such as R-CNN and Fast R-CNN [66], providing high detection accuracy at the cost of high computational complexity [206]. In the HTP field, for example, a model called SlypNet [128] uses variants of two-stages detectors, i.e., Mask R-CNN and U-Net, to perform wheat detection. To overcome the limitations of two-stage detectors, faster (and more accurate) single-stage detectors, such as YOLO and its successors [164],[24], have also been largely employed over recent years. These models improve the ability to extract local features from images and reduce the background detection error rate [112]. For example, FlowerPhenoNet [43] was used to identify flowers and investigate their location within images. Moreover, the authors in [24] compared the effectiveness of YOLOv5 architectures in performing node, fruit, and flower detection on tomato plants. Lastly, the results achieved by integrating ML and DL also led to a comparison of these types of approaches. As

an example, the authors in [218] compared image-based RSA phenotyping methods using several ML and DL algorithms, specifically k-means, naïve Bayes, random forest, shallow neural networks, and deep neural networks, showing that the latter achieved the highest accuracy (about 86%) on a dataset of 617 root images from mature alfalfa plants.

2.3.1 Multi-Stage Detectors

A diverse range of applications harness the power of deep architectures, particularly convolutional neural networks (CNNs). In this domain, two main architectural approaches have gained prominence for detection tasks: two-stage detectors, exemplified by R-CNN and Fast R-CNN [66], and single-stage detectors, represented by YOLO and it is successive versions [164]. In the realm of High-Throughput Phenotyping (HTP), SlypNet [128] adopts an alternative methodology by employing tailored versions of two-stage detectors, such as Mask R-CNN and U-Net, to effectively carry out wheat detection tasks. The objective of object detection is to detect and localize all objects within an image while determining their respective classes and likelihoods [67].

Two-stage detectors adhere to a standardized approach: initiating with a region proposal network (RPN) to generate initial object proposals, followed by a dedicated per-region head for classification and refinement. For instance, Faster R-CNN [167], [76] utilizes two fully connected layers as the RoI (Region of Interest) heads. Cascade R-CNN [22] advances this method by employing three sequential stages of Faster R-CNN, each with distinct positive thresholds to prioritize localization accuracy in later stages.

Faster Region-based Convolutional Neural Network (Faster R-CNN) emerges as a widely employed method for small object recognition, particularly in field conditions, as evidenced by several studies [75], [154], [206], [205]. [75] employed Faster R-CNN to detect apples on untrained trees, achieving a success rate of 90.8% on 103 images under natural lighting conditions. [206] introduced an enhanced Faster R-CNN model with an Attention Mechanism to detect young tomato fruits against near-color backgrounds. While this approach addressed recognition challenges posed by similar background colors, it focused solely on one complex scenario—near color backgrounds—neglecting other factors like illumination, occlusion, and overlap in

diverse environments. [205] enhanced Faster R-CNN by integrating an attention mechanism, enabling the detection of sweet potato leaves in natural environments with an average precision of 95.7%. [154] developed a deep learning method for strawberry instance segmentation, using a novel architecture based on Mask R-CNN backbone and Mask networks. Their approach achieved competitive results, with a mean average precision (AP) of 43.85%, slightly below the original Mask R-CNN 45.36% on a test set of 200 images.

These studies collectively demonstrate the effectiveness of Faster R-CNN in fruit detection tasks. Despite these advancements, two-stage detectors maintain an edge in accuracy across various scenarios. However, current implementations of two-stage detectors often rely on a relatively weak RPN that emphasizes recall over precision, resulting in a large number of proposals without leveraging proposal scores during testing. This abundance of proposals slows down the system, and the recall-oriented RPN lacks the straightforward probabilistic interpretation characteristic of single-stage detectors.

2.3.2 Single-Stage Detectors

To overcome the limitations of two-stage detectors, there has been a significant shift towards faster and more accurate single-stage detectors, notably YOLO and it is subsequent iterations[164],[24].

Numerous studies have favoured YOLO-based algorithms due to their superior precision and faster detection speeds [206]. YOLO and it is successors have been instrumental in detecting various agricultural products such as cucumbers [16], kiwifruit [190], grapes [177], cherries[61], diverse apple varieties [54], and tomatoes[206], [98], [125]. These models excel in extracting local features from images and reducing background detection errors [112]. For instance, researchers utilized YOLO variants to count sorghum heads from drone imagery, achieving an average precision of 95% [143]. In essence, the models derived from the YOLO family strike an optimal balance between model complexity and real-time performance, enabling the capture of intricate patterns and relationships while remaining deployable on resource-constrained hardware, such as terrestrial Vehicles and Unmanned Aerial Vehicles (UAVs).

Over time, YOLO models have advanced towards greater density, resulting in improved accuracy, albeit often at the expense of detection speed. Previous studies primarily focused on YOLOv3, with adaptations tailored to specific challenges. [114] introduced YOLO-Tomato, achieving a 94.58% mAP on 609 images, enhanced further by [206] to 96.41% using dense backbone connections. [99] introduced YOLO-DenseNet and YOLO-MixNet, with YOLO-MixNet reaching an mAP of 98.40%. However, YOLOv3 was surpassed by YOLOv4, as shown by [175], achieving an 81.28% mAP on 2000 images. They highlighted this trend, demonstrating that YOLOv4 outperformed YOLOv3 in mean average precision (mAP) despite slower inference times. Subsequent enhancements, exemplified by YOLO-Tomato by [114] and further refinements by [115] as well as [206], have significantly boosted accuracy while maintaining efficient inference times. [237] attained a 94.44% mAP on 1698 tomato images, while [173] reached a 96.29% mAP on 12,000 tomato images with diseases. In [129], replacing CSP modules with a lightweight version and employing CARAFE resulted in an 82.8% mAP. Additionally, YOLOv3 has served as a cornerstone for models like YOLODenseNet and YOLOMixNet, customized for automated robotic platforms, achieving high mAPs alongside rapid detection speeds.

By increasing the density of Yolo models, [24] explored YOLOv5 effectiveness in identifying flowers, fruits, and nodes, reducing false negatives and accurately labeling missed objects. [171] introduced YOLOv5-4D for object detection and tracking, reaching a 74.8% mAP for tomato cluster counting. [108] improved YOLOv5s with a stepwise partial network and efficient loss function, obtaining the mAP by 0.66%. [163] employed a modified backbone in YOLOv5, raising a 94.10% mAP on a dataset focusing on tomato virus diseases. [230] proposed THYOLO, reducing computational costs by 84.15%. [208] developed SM-YOLOv5 with MobileNetV3-Large, achieving a 98.80% mAP. [233] introduced YOLOXMOB and YOLOXPC for small target identification, surpassing the bare network performance. The introduction of the DSE-YOLO model aimed to address the challenges of multi-stage fruit detection by enhancing small fruit detection accuracy, achieving an mAP of 86.58% [209]. However, [85] proposed the YOLOv5s-MBLS algorithm, which outperformed the original YOLOv5 in complex strawberry planting scenarios, increasing mean average precision by 1.6% and reducing the model size by 21.9%. Another advanced model introduced by [33] incorporated GhostConv, rotation operator, block convolution attention module, and Content-Aware ReAssembly of Features, achieving an mAP50% of 94.7 for real-time strawberry disease control. [79] presented YOLOv5s-

Straw, a custom object detection model tailored for outdoor strawberry detection, reaching the highest mAP of 80.3% under identical conditions compared to other models. To further enhance strawberry recognition during harvest, [27] introduced a real-time and precise detection approach based on the YOLOv7 model with an mAP of 0.89% and an F1 score of 0.92%. Building upon this, [49] developed the DSW-YOLO model, an extension of YOLOv7, which marked a 5.0% improvement in precision, 1.7% in recall, and a 2.2% increase in mAP compared to YOLOv7.

Lastly, [222] enhanced YOLOv8 for tomato harvesting, improving feature extraction and recognition accuracy with a 93.4% mAP while reducing parameters.

2.3.3 Root Systems Architectures (RSAs)

The root plays a crucial role in anchoring the plant and absorbing water and nutrients from the soil, which are essential for its growth. The phenotyping of plant root traits through 2D and 3D imaging techniques is becoming increasingly significant in agriculture to improve the breeding of superior cultivars. But Root system architectures (RSAs) are difficult to observe directly due to being naturally covered by soil [141]. Consequently, specific non-destructive phenotyping methods have been developed, such as using transparent agar or germination papers. These methods have proven efficient, particularly at early growth stages, but they require the root system to grow in artificial conditions[159]. Another viable approach is X-ray computed tomography, which allows visualization of the root system in natural soil[155]. However, this technology is expensive and challenging to deploy directly in the field.

Once images of the Root System Architecture (RSA) have been gathered, they must be segmented to detect the roots. However, this segmentation step is challenging due to the complex nature of RSA and the low contrast between soil particles and roots. To address these challenges, several tools have been developed. While analyzing mini rhizotron (MR) images remains complex, software like RootFly [229] and rhizoTrak [140] facilitate the measurement of root lengths and widths by allowing users to semi-manually trace roots on the images. Manual root identification is tedious, time-consuming, and requires skilled annotators, increasing the demand for semi-automatic software tools. These tools typically begin with the semantic segmentation or identification of roots or related structures such as nodules from RGB images.

While Convolutional Neural Network (CNN) architectures have been explored for root image analyses for over a decade [161], programs such as DART [100], SmartRoot [117], EZ-Rhizo [7], DIRT [21], GiA-Roots [62], RootNav [161], and GT-RootS [17] are generally effective at detecting and analyzing root systems and traits within the specific environments for which they were designed.

For instance, the GLO-Roots framework uses feature-based image analysis techniques, including local pattern recognition, global shape analysis, and directionality analysis, to identify and extract root system characteristics, also considering gene reporters and soil moisture [166]. GT-Roots [17], another semi-automated tool, applies a processing pipeline to each image that starts by extracting a Region of Interest (RoI), converting the original image into grayscale, performing adaptive thresholding, and finally applying a morphological operator to enhance the results. GT-Roots allows for a semi- or fully-automated pipeline, where the operator can manually intervene at each intermediate processing step. GiA-Roots [62] first performs image pre-processing via rotation, cropping, and scaling. The user then selects relevant root system traits from a set of 19 possible choices used on the segmented image to extract the root system. SaRIA [148] provides a semi-automated environment for RSA segmentation and calculating phenotypic features of the RSA, with a pipeline that includes cropping, despeckling, smoothing, and inversion of image intensity, followed by adaptive image thresholding and morphological filtering to improve root quality. Root skeletons are computed, and RSA features are extracted from a list of 44 root traits using pixel-wise computation.

While these tools perform well with a limited number of images, they may be inadequate for processing large datasets due to the required human intervention. Additionally, fixed processing pipelines often lack generalization capabilities. To address these limitations, deep learning-based tools have been developed. One notable tool is SegRoot [207], which provides a binary mask of root (white pixels) and no-root (black pixels) starting from an RSA image. SegRoot is based on a modification of SegNet [10] and uses standard CNN blocks in the encoder, with unpooling layers in the decoder for non-linear upsampling. The primary difference between SegRoot and SegNet is the loss function, a modification of the Dice coefficient [134]. Another tool, DeepLabv3+ [181], uses a U-shaped encoder based on Xception as its backbone. Additionally, an approach has been proposed that predicts two parameters representing the vertical and horizontal centroids of root distribution to reveal the phenotypic diversity of root distribution [193].

Chapter 3

Methodology

3.1 Overview of the Research Type

The integration of qualitative and quantitative methods provides a holistic understanding of the complex interplay between hardware, software, and algorithms in high-throughput plant phenotyping. This comprehensive approach aims to offer valuable insights and practical recommendations for enhancing current methodologies, ultimately contributing to the global effort to increase food production leading up to 2050.

3.2 Description of the Design Framework

The design framework of this study begins with a systematic examination of the impact of hardware and software factors used in plant phenotyping. This framework is crucial for understanding the methods employed and the rationale behind each step to achieve the research objectives.

The primary goal is to enhance the efficiency of detecting plant traits and improve phenotypic outcomes by integrating deep learning models with innovative computational methods.

The design framework includes the following components:

- Selection and systematic analysis of recent articles following the PRISMA protocol
- Selection and implementation of hardware tools for image acquisition techniques (platforms, sensors, and required tools)
- Ethical considerations and data integrity measures
- Rigorous data management and analysis methodology
- Development and optimization of deep learning models

A mixed-methods approach was chosen for this study to leverage both qualitative and quantitative data. This approach allows for a comprehensive analysis of high-throughput plant phenotyping, providing a deeper understanding of the hardware, software, and algorithms involved.

All collected data were managed following ethical guidelines to ensure confidentiality and integrity. The data were analyzed using advanced computational tools and algorithms. Deep learning models were developed and optimized to identify morphological traits in plants. A rigorous method was employed to classify and detect inherent features in the dataset. Several challenges were encountered, including the complexity of data integration and model optimization. These challenges were addressed through repeated refinement of models and rigorous testing procedures to ensure accuracy and reliability.

3.3 Description of the Datasets

The datasets used in this study are essential for the precise detection of flowers, fruits, and nodes on tomato and strawberry plants, and the detection of barly roots. These datasets were selected because they play a crucial role in addressing significant agricultural challenges specific to Italy's tomato, strawberry and barly production.

Italy is renowned for its tomato and strawberry cultivation, which are not only iconic but also economically important crops. The datasets chosen for this research are particularly relevant due to their comprehensive nature, providing detailed information necessary for developing accurate detection algorithms. These algorithms are essential for optimizing agricultural practices, such as monitoring plant health,

improving yield estimation, and implementing targeted interventions to enhance crop productivity.

By leveraging these datasets, this research aims to contribute to the advancement of agricultural technology and sustainable farming practices. The detailed analysis of flowers, fruits, and nodes on tomato and strawberry plants will provide valuable insights into crop development and health, ultimately supporting efforts to overcome challenges in agricultural production and ensure food security in Italy and beyond.

3.3.1 Structure and type of data, data collection techniques

The datasets for this research were sourced from Metapontum Agrobios Research Centre located in Metaponto, Southern Italy, utilizing their advanced phenotyping platform (PhenoLab: High Throughput Plant Phenomics Platform) located in a greenhouse.

This sophisticated infrastructure includes:

- An automated belt conveyor system tailored for potted plants, featuring an advanced tracking mechanism utilizing barcodes and RFID technology for precise plant identification.
- Four strategically positioned sequential camera stations equipped to capture comprehensive 3D images of plants using near-infrared (NIR), ultraviolet (UV), visible light (RGB), and a specialized NIR camera dedicated to root imaging.
- An automated watering system supplemented by a weight measurement station to ensure meticulous irrigation control.
- An integrated ICT infrastructure meticulously designed to facilitate seamless data acquisition, management, and processing.

As shown in Figure 3.1, this platform enables researchers to conduct quantitative, non-destructive analyses of diverse crops and model plants in high-throughput environments. Each plant is sequentially imaged across multiple Scanalyzer3D camera units, using a spectrum of wavelengths to generate a wealth of reproducible and insightful data points that elucidate various aspects of plant development. For

Fig. 3.1 The high-throughput plant phenomics data collection platform (HTP)(A) contains the plant storage system with conveyor belts that carry the plants to the imaging chambers. The background of the image (B) contains the imaging chambers, which are for, from left to right (the actual direction of plant travel), soil NIR, fluorescence, visible light and plant NIR imaging.

example, tomato plants are effortlessly guided to the imaging chamber, where three distinct views are captured: one from above and two lateral views, each positioned at a 90-degree angle. The platform is situated in a glasshouse, providing semicontrolled conditions, with environmental variables measured via a network of nine sensor nodes, including PAR, temperature, relative humidity, and CO2.

The datasets for this research were sourced from Metapontum Agrobios Research Centre located in Metaponto, Southern Italy, utilizing their advanced phenotyping platform (PhenoLab: High Throughput Plant Phenomics Platform) located in a greenhouse. Tomato, strawberry and barly plants are automatically transported to a specialized imaging chamber for observation and analysis. Within this chamber, each plant is photographed from multiple angles: one image is taken directly above (Top View, TV), while two additional images are captured from the sides (Side View, SV), each at a 90-degree angle from the other. To ensure optimal visibility and clarity, the plants are illuminated using standard fluorescence light tubes emitting a cool daylight spectrum (35 W/865). These light sources provide consistent and uniform lighting conditions for imaging purposes. The side-view (SV) image dataset used in this research is specifically chosen for its ability to highlight critical features,

such as nodes, from a unique angle, enhancing their visibility. This perspective is essential for accurate identification and analysis, as it captures features that may not be as prominent in other viewpoints, such as the nodes in tomato plants. The data consists of high-resolution images in PNG format, capturing various aspects of tomato, strawberry, and barley plants. Each dataset includes images focused on different plant parts, such as fruit, flower, node, and root. These images represent phenotypic traits and are organized into directories based on plant species and parts, providing a comprehensive visual dataset. The visual data includes detailed images of tomato fruits, flowers, nodes; strawberry fruits, flowers; and barley roots. These images capture variables such as color, texture, size, and shape, enabling detailed phenotypic analysis. The comprehensive nature of these images allows for the extraction of quantitative data on plant health, growth patterns, and other vital phenotypic characteristics. The datasets from the ALSIA Agricultural Research Center provide a robust and detailed visual representation of the plants, facilitating thorough phenotypic analysis and supporting the research objectives with high-quality, consistent visual data of critical plant traits.

3.3.2 Instruments used for data collection

The imaging process is facilitated by a Basler Scout camera, specifically designed for such applications. Additionally, a specialized RGB camera featuring a high-quality SONY ICX274 CCD sensor is employed. This sensor includes a KAI 2093 sensor with dimensions of 1624×1234 pixels, a global shutter mechanism, and a resolution of approximately 2.11 megapixels. Each pixel on the sensor measures $8.50 \times 6.80\,\mu$m, ensuring detailed and precise image capture.

Furthermore, the lens attached to the RGB camera is tailored to meet the requirements of the imaging setup. It conforms to a 2/3 format and utilizes a C mount. The lens has a variable focal length ranging from 12.5 to 75.0 mm, offering flexibility in capturing images at different distances and perspectives. Its maximum aperture of 1:1.8 allows for excellent light transmission, particularly in low-light conditions. The lens is driven by a type 1 motor, powered at 6 V with a maximum current draw of 36 mA, ensuring smooth and reliable operation during image acquisition.

3.4 Preprocessing and Data Cleaning Procedures

Effective data analysis hinges on the quality and preparation of the data. This section outlines the steps taken to preprocess and clean the raw data collected for this study. The goal of preprocessing is to transform raw data into a format that is suitable for further analysis, ensuring accuracy and completeness.

3.4.1 Data Cleaning

The raw dataset underwent rigorous cleaning procedures to address various issues such as missing values, outliers, and inconsistencies. During this stage, images lacking the desired plant traits (desired classes) were filtered out using Python code. These images depicted plants in early growth stages without flowers or fruits. Additionally, outliers were identified and manually removed from the dataset.

3.4.2 Handling Categorical Data

Handling categorical data in the context of image data involves encoding labels to facilitate model training. For this study, each image in the dataset was labeled according to predefined categories corresponding to different plant species or growth stages. This labeling process was crucial for establishing ground truth annotations, ensuring that the neural network could learn to recognize and differentiate between specific features of interest within each image. The labels were encoded using a categorical encoding scheme, where each unique category was assigned a numerical identifier. This approach not only prepared the data for model training but also standardized the input format, enabling efficient handling of categorical information throughout the neural network architecture.

3.4.3 Tools and software used for preprocessing

For preprocessing the dataset, the PlantCV software, a robust tool designed specifically for plant phenotyping, was employed. PlantCV was utilized to filter and scale the images, ensuring that only high-quality data was used for further analysis. The software's advanced image processing capabilities allowed us to effectively manage

and enhance the dataset by performing tasks such as noise reduction, color correction, and morphological transformations. This preprocessing step was crucial to normalize the images and remove any extraneous elements, thereby optimizing the dataset for subsequent training and analysis.

3.5 Software and tools employed

After the data collection phase, domain experts engaged in the meticulous annotation of images using the Computer Vision Annotation Tool (CVAT). CVAT was chosen for its robust capabilities in annotating complex visual data, allowing experts to label specific features such as plant traits or growth stages with precision and consistency. This tool facilitated the creation of annotated datasets that serve as ground truth for subsequent machine/deep learning tasks. Its versatility in handling various annotation types, including bounding boxes, polygons, and keypoints, ensured that the annotated data met the stringent requirements of the research project. Moreover, CVAT's collaborative features enabled seamless teamwork among annotators, ensuring efficient annotation workflows and maintaining annotation quality across the dataset.

After preprocessing data, the machine used for the experiments was based on a Windows 11 operating system, equipped with an NVIDIA GeForce RTX 3080 GPU with 10 GB of RAM and an Intel Core i9-11900HK CPU with 32 GB of RAM. The framework used for deep learning was based on the Ultralytics package and PyTorch 1.11.0.

3.6 Model Evaluation

The evaluation metrics used in this study are primarily four: *precision* (P), *recall* (R), F1-score, and *mean average precision* (mAP). To briefly review, precision and recall are calculated as follows. For simplicity, the *binary* case will be considered, involving a classification problem with two classes: one labeled as *positives* and the other as *negatives*. The formulas for P and R are given by the following equations.

$$P = \frac{TP}{TP+FP} \tag{3.1}$$

$$R = \frac{TP}{TP+FN} \tag{3.2}$$

In the previous equations:

- **TP** are the *true positives*, that is, the instances correctly identified by the model as samples of the *positive* class.
- **TN** are the *true negatives*, that is, the instances correctly identified by the model as samples of the *negative* class.
- **FP** are the *false positives*, that is, the instances incorrectly identified by the model as samples of the positive class.
- **FN** are the *false negatives*, that is, the instances incorrectly identified by the model as samples of the negative class.

The F1-score can be derived from P and R as follows:

$$F1 = 2\frac{P \cdot R}{P+R} \tag{3.3}$$

In a multi-class problem like the one under investigation, these metrics are calculated for each class pair and then averaged, weighted by the number of samples in each class. To evaluate the mAP, the concept of *Intersection over Union* (IoU) is introduced, defined as:

$$IoU = \frac{A_O}{A_U} \tag{3.4}$$

In Equation 3.4, A_O represents the overlap area between the ground truth and the predicted box, while A_U denotes the union of these areas. The IoU ranges from 0

to 1 and is directly proportional to the overlap between the predicted box and the ground truth. An IoU of 1 indicates complete overlap, whereas an IoU of 0 indicates no overlap.

Typically, an IoU threshold of 0.5 is used to confirm a detection. Using this threshold, the *average precision* (AP) can be calculated as the area under the *precision-recall* curve at the specified IoU threshold. Since AP is computed for each class, the *mAP* is the mean of the AP values across all classes. In the experiments, *mAP* was evaluated using two different thresholds:

- $mAP - 0.5$, which is the value for the *mAP* computed considering an IoU threshold of 0.5.
- $mAP - 0.5 - 0.95$, which is the value for the *mAP* computed for 10 different IoU threshold ranging from 0.5 to 0.95 at a step frequency of 0.05, and then averaged.

Chapter 4

Detection of Tomato Plant Phenotyping Traits Using YOLOv5 Model

4.1 Overview

Tomatoes hold a prominent status in Italy, recognized as a crucial crop among both fruits and vegetables due to their extensive cultivation. Consequently, optimizing tomato production has become a central challenge in digital agriculture. This endeavor is complicated by factors such as quality deterioration due to shading and the need to distinguish between ripe and unripe fruits promptly. Addressing these challenges involves extracting relevant data from images, specifically focusing on phenotyping traits encompassing tomato fruits, flowers, and stem nodes, and meticulously observing them throughout the plant developmental stages. This approach enables continuous evaluation of tomato quality, thereby streamlining efforts to enhance overall production. By analyzing changes in key phenotypic characteristics, stakeholders can gain valuable insights into plant adaptive responses, allowing for refining production strategies in response to evolving conditions.

The utilization of computer vision techniques, particularly when combined with deep convolutional neural networks (CNNs), offers a practical solution to monitor responses to external factors like nutritional inputs (e.g., fertilization) and environmental stressors (e.g., drought and salinity). In cereals, these methods have been

effectively applied to detect pests and diseases [210]. They also allow for rapid identification of plant organs, abiotic stresses, and the ability to separate crops from weeds. Additionally, these techniques can be used to count leaves with visible symptoms, such as those caused by leaf mines from "Tuta absolute" in tomatoes. Infested leaves are distinguishable from healthy ones by differences in shape, color, and pattern. Traits like tomato fruit size and number [132]; [153], stem and internode elongation [113], and flower count and setting [153] are heavily affected by biotic and abiotic stresses. This research aims to improve the detection of such stresses in tomatoes. Previous studies typically focused on using individual models to identify specific traits, like machine learning models for detecting tomato plant nodes [220], or machine and deep learning approaches for identifying tomato fruit ([221]; [144]). An interesting set of applications involves deep architectures, such as convolutional neural networks (CNNs). Specifically, two types of architectures have been used as detectors: two-stage detectors, such as R-CNN and Fast R-CNN [66], and single-stage detectors, such as YOLO and its successors [164]. Both these types have been successfully exploited in phenotyping-oriented scenarios. However, many studies have favoured YOLO-based algorithms due to their high accuracy and faster detection speed [206]. For example, YOLO and its successors have been used for detecting the inter-node length of cucumbers [16], kiwifruit [190], grapes [177], cherries [61], apple varieties [54] and tomatoes ([206], [98], [125]).

[114] introduced a modified version of YOLOv3 called YOLO-Tomato, which incorporates two key changes. First, circular bounding boxes were used instead of rectangular ones to better fit the shape of tomatoes. Second, dense layers were added to the network backbone. These enhancements resulted in 94.58% accuracy under slight occlusion conditions and 90.10% accuracy under severe occlusions on a dataset of 303 tomatoes. Specifically, in [99], two variants of YOLOv3 have been proposed, called YOLODenseNet and YOLOMixNet. The two main differences between the two variants lie in their backbone: YOLODenseNet has been developed using a DenseNet-based backbone, while YOLOMixNet has been based on a mixture of DenseNet and DarkNEt. Still, both architectures have exploited a common set of machine learning techniques, such as image pyramid and complete IoU. Both networks have been tested against a dataset of 485 images for tomato detection, achieving an mAP of 98.3% and 98.4% for YOLODenseNet and YOLOMixNet, respectively, with an average detection speed of 21.88 and 21.1 m/s. A similar set of improvements has been proposed by [206], where dense connections have been

used in the backbone, along with K-means for computing the size of anchor boxes and multiscale training. The model has been tested against the same dataset used by [115], achieving an improved accuracy of 96.41% with a reduced detection time of 20.28 m/s. As for models based on denser architectures, [237], have modified the standard CSPDarkNet53 backbone of YOLOv4 by adding residual layers, achieving an overall average accuracy of 94.44% on a dataset composed of 1698 images of mature and immature tomatoes. Also, [173] have used a modified version of YOLOv4 with two types of layers within the backbone to enhance the receptive field and preserve fine-grain localized information for tomato disease detection. Specifically, the network has been tested against a dataset containing 12000 images of four different plant diseases, that is, early and late blight, Septoria leaf spot, and leaf mold achieving an overall mAP of 96.29%.

However, there has been insufficient emphasis on developing CNN models capable of concurrently recognizing multiple traits of plants, especially for tomato plants. This chapter aims to present advanced CNN models for the single and multi-classification of tomato traits (nodes, flowers, and fruits) simultaneously, by implementing both pre-trained and custom-designed algorithms. Additionally, the models were tested with diverse backbones to evaluate how different network layers and configurations influence the outcomes. A dataset consisting of 1683 tomato images was employed to assess the effectiveness of the proposed models. The results indicate that the models achieve notable performance metrics, particularly given the challenges posed by variations in object size, similarity, and color within the input images.

4.2 Methodology of Tomato Plant Phenotyping Detection

Detecting tomato plant traits using single-stage detectors (YOLOv5 model) in this chapter is composed of several components. The first subsection elaborates on the Plant Phenotyping Platform utilized within the proposed methodology. In the second subsection, the image acquisition method will be explained. The third subsection discusses the methods used to prepare the datasets. In the fourth subsection, the developed deep learning algorithms (YOLOv5 architecture) will be discussed. Finally,

the fifth subsection discusses the evaluation metrics used to assess the proposed methods' detection and classification capabilities.

4.2.1 Data Preparation

The side-view (SV) image dataset used in this research is specifically chosen for its ability to highlight critical features, such as nodes, from a unique angle, enhancing their visibility. This perspective is essential for accurate identification and analysis, as it captures features that may not be as prominent in other viewpoints, such as the nodes in tomato plants. The dataset was collected during a stress experiment on various tomato genotypes using the High-Throughput Phenotyping (HTP) platform, a sophisticated system designed for comprehensive plant analysis. In this experiment, tomato plants were grown in pots with a specific sand and peat mixture. To induce drought stress, irrigation water was reduced by 70% over two stress cycles, followed by recovery phases. This setup allowed researchers to observe how different tomato genotypes respond to drought stress conditions.

Over several weeks, RGB images were captured at various time points corresponding to specific stages of the experiment. These images serve as digital phenotypes, providing detailed visual representations of both control plants and those subjected to drought stress. The use of RGB images ensures accurate capture of various plant aspects, including color and morphology. Each image in the dataset has a fixed resolution of 1624×1234 pixels, ensuring consistency. After data collection, domain experts meticulously annotated the images using the Computer Vision Annotation Tool (CVAT), as depicted in Figure. 4.1 This annotation process involved outlining and labeling specific regions of interest within the images, focusing on three key phenotyping traits: flowers, fruits, and nodes.

One notable challenge with the annotated dataset is the inherent imbalance among the provided classes, as shown in Figure. 4.2. There are significantly more annotations for certain classes compared to others, such as nodes having more annotations than flowers or fruits. This class imbalance could potentially bias the model training process towards the over-represented class. To address this issue and ensure a more balanced training process, various data augmentation techniques were employed. These techniques artificially alter the training data to create additional variations, thereby reducing the risk of model bias. Examples include random affine

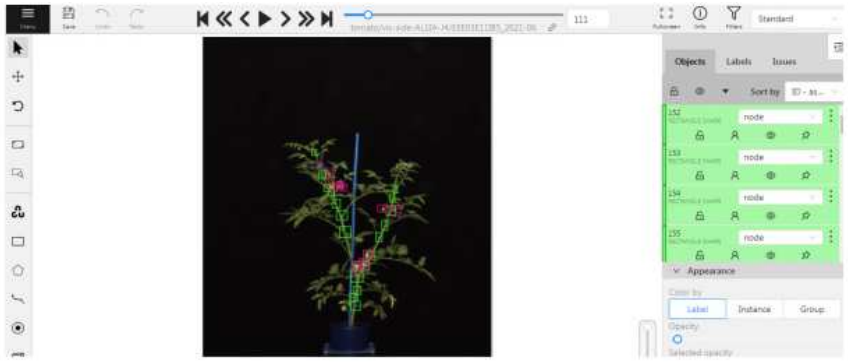


Fig. 4.1 The Graphical User Interface provided by the Computer Vision Annotation Tool (CVAT) allows users to manually insert bounding boxes around relevant objects and subsequently label them with appropriate annotations. In this case, a domain expert labeled examples of flowers, fruits, and nodes.

transforms, which alter the position, scale, rotation, and shear of the images, as well as HSV augmentation and random horizontal splitting. These augmentation methods help diversify the dataset and prevent the model from becoming overly reliant on any specific class or feature, ultimately improving its ability to generalize to new data.

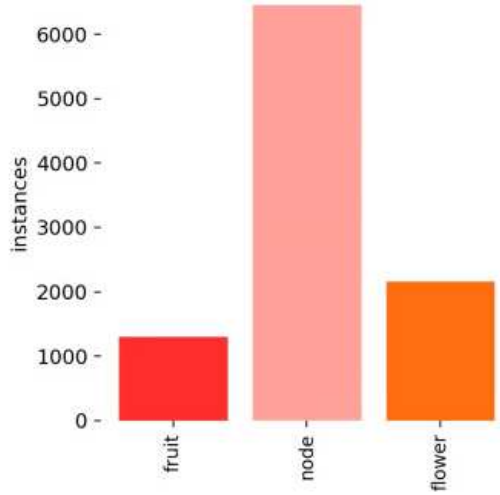


Fig. 4.2 Number of instances per class: Overall, there are 1862 fruit labels, 9276 node labels, and 3111 flower labels. Consequently, the dataset is imbalanced.

The dataset images exhibit notable variation in the appearance of the three classes: nodes, fruits, and flowers, as shown in Figure. 4.3. This variability stems from the inherent dynamism of plant growth. Plants undergo continuous development, resulting in significant differences in size and morphology not only between individual plants

but also within the same plant at different growth stages. Consequently, the bounding boxes used to delineate these classes can vary widely in size, ranging from very small instances to much larger ones. This variance poses a challenge, particularly when identifying nodes during the early stages of plant growth, where they may be exceedingly small and intricately intertwined with other plant structures.

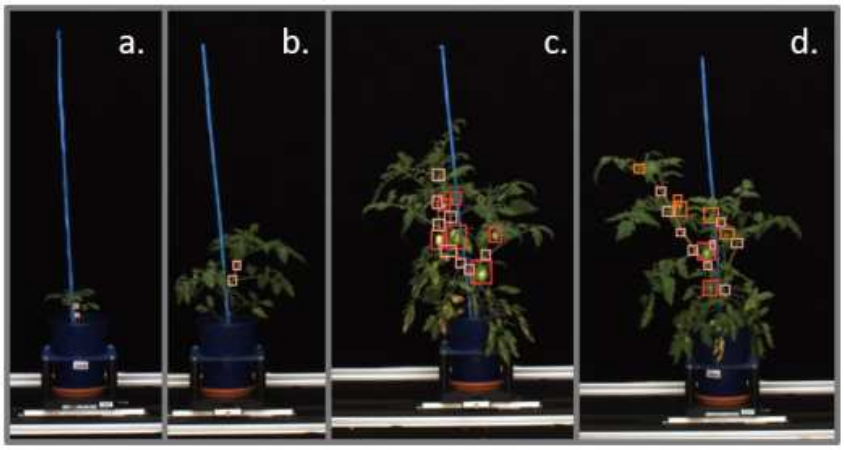


Fig. 4.3 Example of labeled images in the dataset: Pink bounding boxes indicate nodes, orange bounding boxes indicate flowers and red bounding boxes indicate fruits.

Furthermore, the diversity in fruit color within the dataset adds another layer of complexity to the analysis. Some fruits may exhibit a yellow hue, while others may appear green, and notably, there are no instances of red tomatoes among the images. This variation in fruit color reflects natural genetic diversity and environmental influences, such as ripeness and exposure to light. Similarly, the appearance of flowers can vary widely, encompassing a spectrum of shapes, sizes, and colors. Some flowers may be fully bloomed and easily recognizable, while others may be in earlier stages of development or obscured by foliage, making them more challenging to identify. Understanding these nuances in the dataset is crucial for developing accurate and robust analysis algorithms. Researchers must devise methods capable of effectively discerning and characterizing these classes across a wide range of visual attributes, ensuring reliable and consistent results across diverse plant specimens and growth conditions. This highlights the importance of employing sophisticated image processing and machine learning techniques capable of accommodating the variability and complexity inherent in biological datasets.

4.2.2 YOLOv5 Architecture

To comprehensively introduce the architecture employed in this study, it is essential to explore the mechanics of advanced object detectors based on deep Convolutional Neural Networks (CNNs). These detectors function through several critical stages:

- Grid Division: The input image is segmented into a grid of smaller sections.
- Feature Extraction: A backbone CNN processes each grid section to extract pertinent features.
- Global Relationship Modeling: A component called the "neck" integrates the features from various grid sections, capturing global relationships within the image.
- Detection: The "head" of the detector utilizes the integrated relationships and extracted features to generate final detection outcomes, including bounding box coordinates, confidence scores, and class probabilities.

In the realm of object detection, two primary categories of detectors exist two-stage detectors and single-stage detectors. Two-stage detectors follow a sequential approach, first focusing on localizing objects and then on their classification. On the other hand, single-stage detectors amalgamate both localization and classification within a single step. Studies indicate that while two-stage detectors typically deliver marginally superior accuracy, single-stage detectors excel in rapid detection speeds.

The YOLOv5 architecture represents a progression from the YOLO (You Only Look Once) family, falling under the single-stage detectors category. Its framework incorporates the following components:

- Backbone: Employing the advanced CSP-Darknet53, where CSP denotes Cross-Stage-Partial-connections, the backbone module extracts crucial features from the input image.
- Neck: This part integrates SPPF (Spatial Pyramid Pooling - Fast) alongside the latest CSP-PAN (Pyramid Attention Network), refining the extracted features and capturing spatial correlations within the image.

- Head: Utilized across diverse YOLO architectures, the detection head concludes the detection process by producing bounding boxes, confidence scores, and class probabilities.

In YOLO-based architectures, images are divided into an S × S grid, with S being a fixed value, as shown in Figure 4.4. Each grid cell predicts a set of bounding boxes, each accompanied by a confidence score and conditional class probabilities. By combining the confidence scores with the class probability maps, the detector produces the final detections

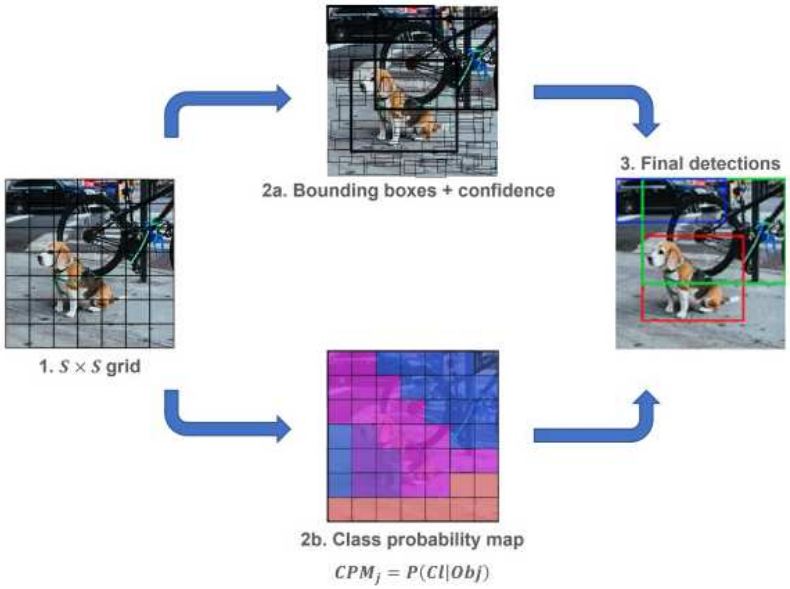


Fig. 4.4 Working principles of YOLO-based architectures. First, the detector divides the image into a grid of S×S cells. Next, for each grid cell, a class probability map and a confidence score are computed for the estimated bounding boxes. Finally, the confidence score is used to determine the final detections.

In this study, six versions of the YOLOv5 architecture were selected for comparison. These models share the same foundational architecture, as described by [191], but differ primarily in their density, which refers to the number of parameters and layers in the network. YOLOv5s represents the smallest and least complex version, while YOLOv5x6 is the most dense. Comparing these versions is crucial for evaluating their achievable accuracy. Typically, denser networks yield better results in terms of evaluation metrics, provided there is sufficient training data. However, denser models may also result in higher inference times, which can be significant, especially

for computationally intensive tasks. This study primarily focuses on accuracy, with future plans to assess inference time performance on a dedicated testbed.

Due to the limited amount of available data, training the entire network from scratch resulted in sub-optimal performance. Therefore, transfer learning was employed: the weights of the neurons in the backbone were frozen, allowing the head to focus on the specific problem while utilizing the knowledge the backbone gained from larger datasets. Each training session involved 300 epochs of transfer learning, with results evaluated in terms of precision, recall, and mAP, as detailed in the next subsection. A standard 60/20/20 split was used for the training, validation, and test sets. Images were resized to 1280 × 1280 pixels during the input stage.

4.3 Experimental Results and Discussion

The study introduces and applies the YOLOv5 architecture, enhancing its performance in tomato detection. Through thorough analysis of a dataset comprising 1683 tomato images, significant improvements are observed compared to previous research efforts. This section is structured into three parts. Firstly, the data are trained and evaluated using the base YOLOv5 architectures, employing transfer learning. Subsequently, the second part explores enhancements employing two techniques: Test-Time Augmentation (TTA) and model ensembling, aimed at refining results obtained through transfer learning. Finally, the third part investigates the impact of different backbones on detection efficacy, analyzing various network layers and configurations to comprehend their influence on overall performance.

4.3.1 Transfer Learning

Let us kick off this discussion by delving into the outcomes derived from employing alternative methodologies, specifically focusing on the refined YOLOv5 architectures trained via transfer learning, as illustrated in Table 4.1.

Table 4.1 Results achieved on tomato recognition. As expected, the wider architectures, that is, YOLOv5l6 and YOLOv5x6, provide the best results.

Class	Model	TP	M	B-FP	B-FN
Fruit	YOLOv5s	64%	1%	7%	35%
	YOLOv5m	73%	3%	7%	24%
	YOLOv5l	75%	3%	7%	23%
	YOLOv5l6	76%	3%	7%	22%
	YOLOv5x	75%	4%	8%	22%
	YOLOv5x6	78%	4%	8%	19%
Nodes	YOLOv5s	50%	0%	48%	50%
	YOLOv5m	64%	0%	54%	36%
	YOLOv5l	69%	0%	59%	31%
	YOLOv5l6	66%	0%	58%	34%
	YOLOv5x	69%	0%	59%	31%
	YOLOv5x6	68%	0%	59%	31%
Flowers	YOLOv5s	48%	1%	45%	51%
	YOLOv5m	56%	1%	39%	44%
	YOLOv5l	59%	2%	34%	38%
	YOLOv5l6	64%	2%	34%	34%
	YOLOv5x	59%	1%	33%	39%
	YOLOv5x6	64%	2%	33%	34%

Within Table 4.1, TP embodies true positives, instances where the network accurately identifies labeled objects. Meanwhile, M represents mismatches, highlighting instances where the network misclassifies labeled objects. B-FP stands for background false positives, indicating instances where the model erroneously identifies boxes without corresponding labels from domain experts. Lastly, B-FN represents background false negatives, elucidating labeled bounding boxes that evade detection by the network.

An initial observation is warranted regarding the consistently low rate of mismatches across all three categories. This trend underscores the proficiency of the YOLOv5-based detector on our dataset in effectively analyzing images and distinguishing between fruits, flowers, and nodes. It is noteworthy that within our dataset,

the visual similarities between fruits and flowers can be quite pronounced, presenting challenges even for experienced human annotators in accurately labeling them. However, this is not the case for nodes, as their visual characteristics differ notably from the other two categories. The superior performance of denser networks becomes evident, particularly with YOLOv5l6 and YOLOv5x6 models, which exhibit a slight edge in overall accuracy. Furthermore, while fruit recognition demonstrates minimal false positives and negatives, both nodes and flowers exhibit significantly higher error rates. This discrepancy can be attributed to the following factors.

- Nodes exhibit elevated levels of B-FP due to the labeling by domain experts, who focused solely on nodes along the main stem. However, the network struggles to discern between the main stem and its offshoots. Hence, the heightened B-FP values likely stem from the identification of these auxiliary nodes.
- In the case of flowers, pixels representing yellow chlorotic leaf tissue may be erroneously classified as yellow flower pixels, contributing to increased B-FP values.
- The adoption of denser models notably reduces background false negatives. This indicates that these models can grasp intricate visual connections at a higher level of abstraction, effectively characterizing objects.
- Intriguingly, the aforementioned point is bolstered by the fact that denser models exhibit higher B-FP values for nodes. This implies that the model is detecting more nuanced nodes on secondary stems.

The bounding box size plays a significant role in the performance of the YOLO detector, as it impacts both the accuracy of detecting objects and the likelihood of false positives and false negatives. Bounding boxes are used by the detector to locate objects within an image, and their size can influence how well the model detects and classifies those objects. Smaller models, like YOLOv5s, tend to have higher miss rates and false negative (B-FN) rates, particularly for classes with smaller bounding boxes, such as the Nodes and Flowers classes. For instance, the miss rate for the Fruit class is higher in YOLOv5s (36%) compared to the larger YOLOv5x6 (22%). This suggests that smaller models struggle more to localize objects accurately, especially when they are small or densely packed within the image. YOLOv5s, with

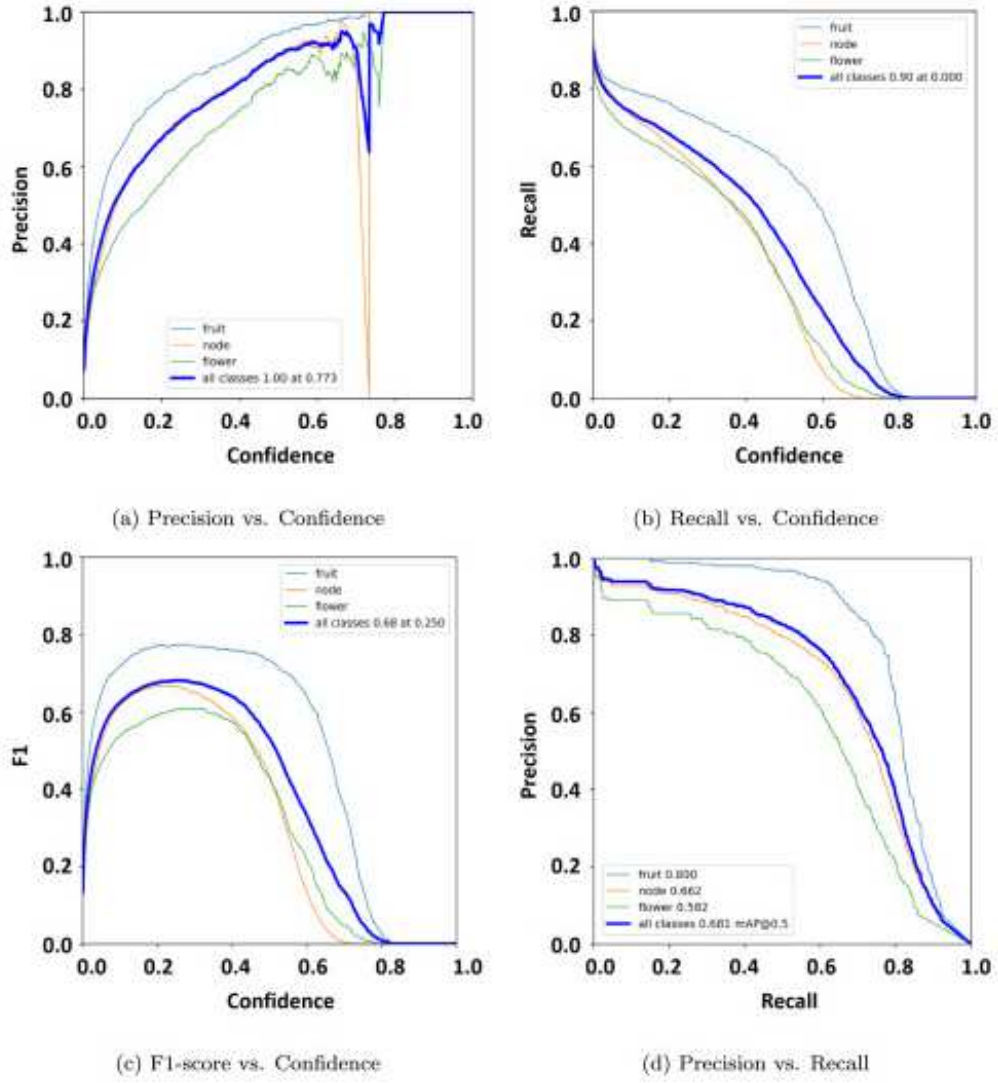
its lower capacity, may lack the resolution or feature representation required to detect smaller objects, leading to more missed detections. The larger architectures, such as YOLOv5x6, are better equipped to handle objects of varying sizes. They achieve better performance in detecting objects with large and small bounding boxes, as seen in the reduced miss rates and B-FN values. For example, YOLOv5x6 reduces the B-FN in the Flowers class from 51% (in YOLOv5s) to 34%. This reduction highlights the model's improved ability to detect smaller bounding boxes, likely due to its higher capacity to capture finer details and features of objects. However, larger models also tend to perform better when it comes to false positives (B-FP), especially for classes with overlapping or complex objects, such as Nodes. For the Nodes class, YOLOv5s has a B-FP rate of 48%, but YOLOv5x6 reduces it to 31%, suggesting that larger models are more effective at avoiding redundant or spurious detections. The reduction in false positives could be attributed to the model's increased ability to differentiate between true objects and background noise, especially when objects are medium-sized or overlapping.

Let us delve deeper into the performance of the top-performing architectures, YOLOv5l6 and YOLOv5x6, by examining additional metrics such as precision, recall, and F1 score. These metrics are visualized in Figure 4.5 and Figure 4.6. In particular, as you traverse these figures clockwise starting from the top left corner, you'll encounter the *Precision/Confidence*, *Recall/Confidence*, *F1-score/Confidence*, and *Precision/Recall* curves for the YOLOv5l6, which is the less dense architecture.

In Figure 4.5, a significant decline in precision occurs around a confidence score of 0.7, suggesting a critical threshold for the network in node identification. This indicates inherent uncertainty in the process. Interestingly, this phenomenon is absent in YOLOv5x6, as illustrated in Figure 4.6, suggesting that denser models can mitigate confidence-related constraints observed in smaller models. Additionally, examining both figures reveals that the F1-score peaks at a confidence level of approximately 0.4.

4.3.2 TTA and Model Ensembling

The performance of models trained through transfer learning can be improved by utilising two effective techniques: Test-Time Augmentation (TTA) and model ensembling. TTA involves applying data augmentation methods during the testing



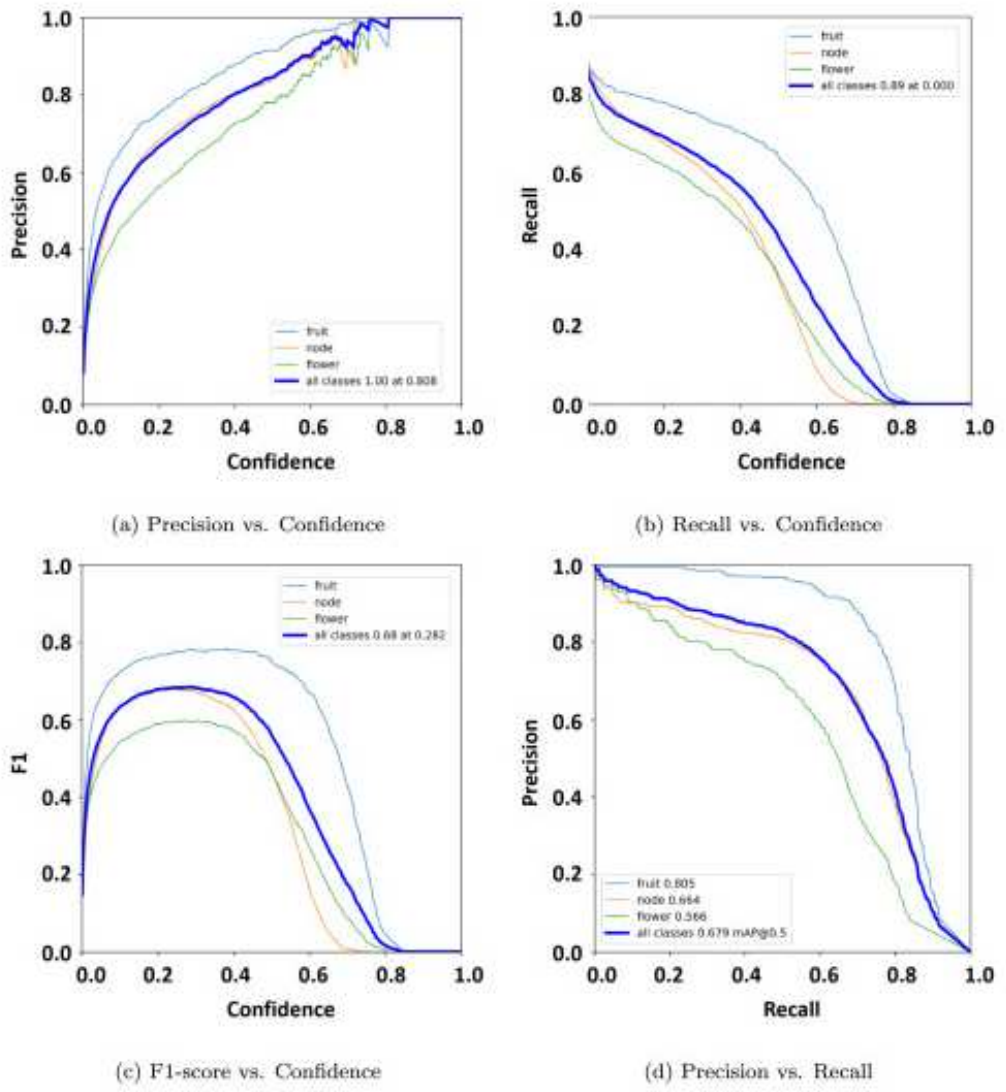
(a) Precision vs. Confidence

(b) Recall vs. Confidence

(c) F1-score vs. Confidence

(d) Precision vs. Recall

Fig. 4.5 Precision, recall and F1 score achieved by the YOLOv5l6 architecture after 300 epochs of training.



(a) Precision vs. Confidence

(b) Recall vs. Confidence

(c) F1-score vs. Confidence

(d) Precision vs. Recall

Fig. 4.6 Precision, recall and F1 score achieved by the YOLOv5x6 architecture after 300 epochs of training.

phase, wherein multiple slightly modified versions of each test image are generated and fed into the model for prediction. The outcome is then derived from the ensemble of these predictions. On the other hand, model ensembling exemplifies ensemble learning, as described by [93]. It combines predictions from diverse models, often through consensus, to produce a comprehensive prediction.

To delve into the analysis, let us examine the outcomes obtained through the ensemble of YOLOv5x and YOLOv5x6, denoted as YOLOv5x+6. The validation results yielded by this ensemble are presented in Table 4.2.

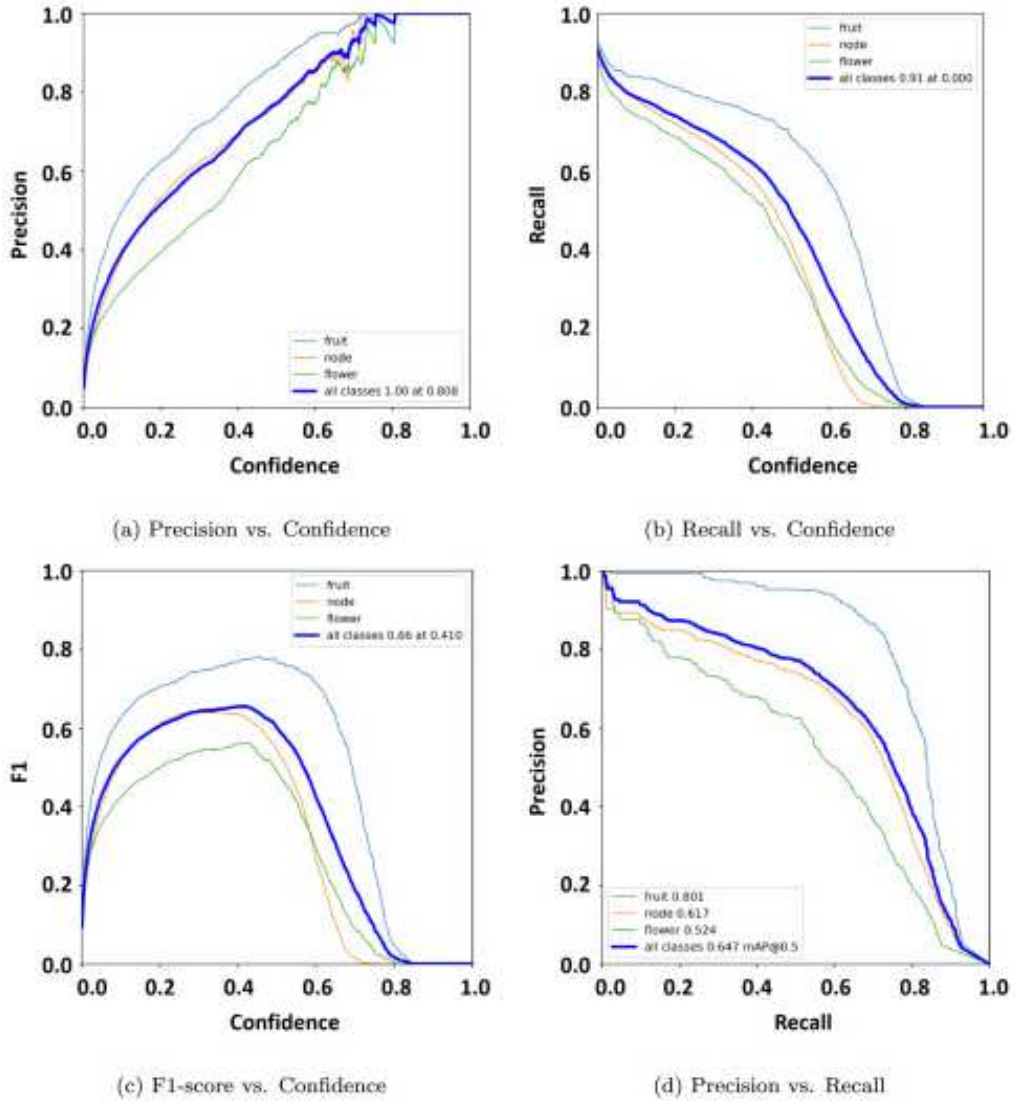
Table 4.2 Results achieved on tomato recognition using the ensemble provided by YOLOv5x and YOLOv5x6, namely YOLOv5x+6. The overall improvement to the bare versions of the architecture is about 3% for fruit, 6% for nodes, and 7% for flowers.

Class	**TP**	M	**B-FP**	**B-FN**
Fruit number	81%	4%	9%	15%
Nodes	75%	0%	56%	25%
Flowers	71%	2%	27%	34%

The impact of ensembling is evident in the results. To illustrate, let us compare the performance of the ensemble model with one of the top-performing single models, YOLOv5x6. The ensemble enhances fruit detection by 3%, node detection by 7%, and flower detection by 7%. Regarding mismatches, there is no significant improvement for flowers; however, there is a notable reduction in false negatives for fruit and nodes, decreased by 4% and 6%, respectively. False positives show interesting trends as well. While the ensemble model yields comparable results for fruits, there is a marked improvement for both nodes and flowers. It is important to note that the elevated values for these classes may stem from incomplete labeling by domain experts or visual artefacts related to overlapping.

Figure 4.7 presents the precision, recall, and F1 score achieved by the ensemble. As anticipated, precision significantly improves with the use of ensembling. Additionally, it is worth noting that the peak F1-score occurs at approximately 0.5 confidence, indicating enhanced overall accuracy in item detection by the resulting model.

Let us consider another ensemble, referred to as YOLOv5+, which includes models YOLOv5l, YOLOv5l6, YOLOv5x, and YOLOv5x6. The results for YOLOv5+ are reported in Table 4.3.



(a) Precision vs. Confidence

(b) Recall vs. Confidence

(c) F1-score vs. Confidence

(d) Precision vs. Recall

Fig. 4.7 Precision, recall and F1 score achieved by YOLOv5x+6.

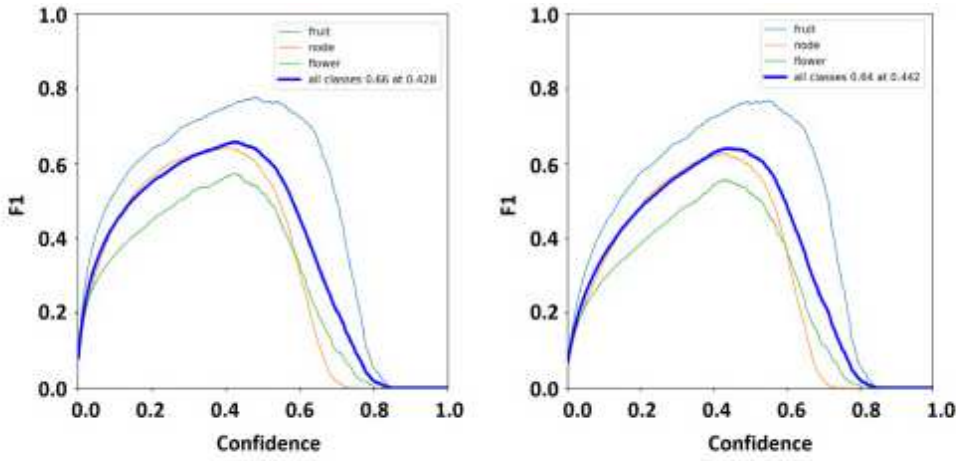
Additionally, Table 4.4 presents the results achieved by YOLOv5+ with Test Time Augmentation (TTA). Finally, Fig. 4.8 displays the F1 scores for YOLOv5+ in Fig. 4.8 (a) and YOLOv5+ with TTA in Fig. 4.8 (b).

Table 4.3 Results achieved on tomato recognition using YOLOv5+. If compared to the ensemble of YOLOv5x and YOLOv5x6, the overall results are improved of about 1% in terms of fruit detection, and 5% for node and flowers detection.

Class	TP	M	B-FP	B-FN
Fruit number	82%	4%	10%	13%
Nodes	80%	0%	55%	20%
Flowers	76%	2%	21%	35%

Table 4.4 Results achieved on tomato recognition using YOLOv5+TTA. If compared to the same model without Test Time Augmentation, the overall results are improved of about 1% for fruit detection and 3% for nodes and flowers detection.

Class	TP	M	B-FP	B-FN
Fruit number	82%	7%	9%	11%
Nodes	83%	0%	58%	17%
Flowers	79%	3%	33%	18%



(a) F1-score - YOLOv5+ (b) F1-score - YOLOv5+TTA

Fig. 4.8 F1 scores achieved by YOLOv5+ and YOLOv5+TTA.

The comparison indicates that YOLOv5+ achieves superior results compared to all other models and ensembles. Specifically, YOLOv5+ improves true positive

rates in fruit detection by 1%, and in node and flower detection by 5% without TTA and 8% with TTA. However, this enhancement comes at the cost of slightly higher mismatches and false positives, especially with TTA. It is important to note that these false positives mainly involve nodes on secondary stems. From a visual perspective, these detections are still correct as the models accurately recognize nodes regardless of their location in the image. Additionally, the overall number of false negatives is significantly reduced, particularly with YOLOv5+TTA, indicating the model improved ability to handle smaller bounding boxes and minimize missed detections. Regarding F1 scores, the trend observed with the YOLOv5x+ ensemble is confirmed, and notably with YOLOv5+TTA, it is evident that the network can effectively identify objects within the tomato plant.

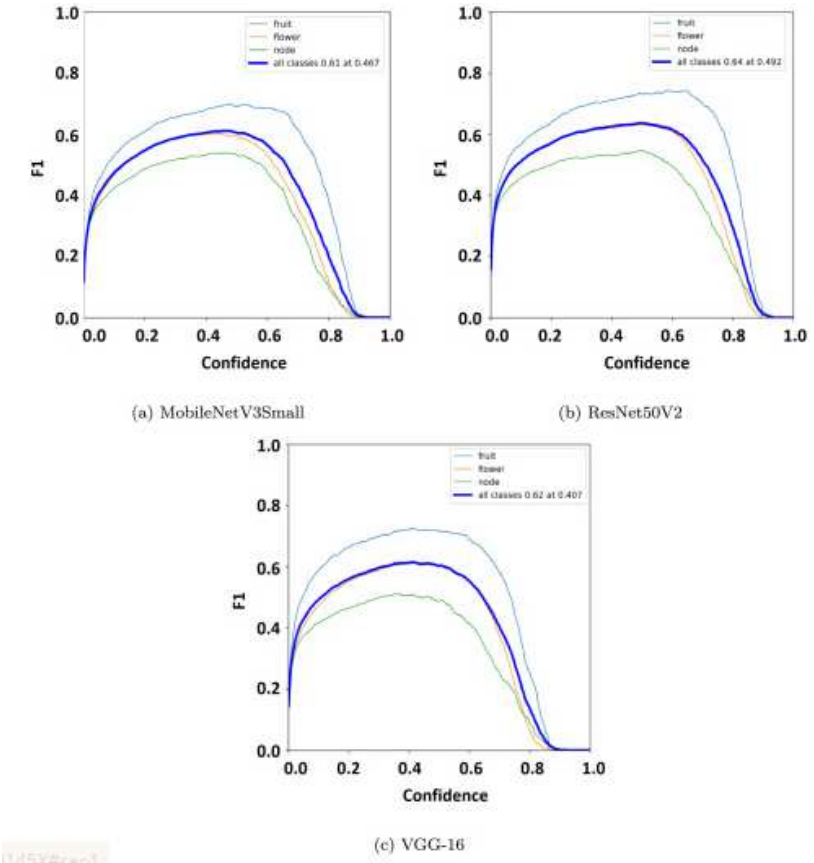
4.3.3 Different Backbones

The final experiment on the dataset involves using different backbones to evaluate the impact of various network layers and configurations on the results. This state-of-the-art comparison was conducted because it is not feasible to directly feed images from the dataset into the models reviewed in the related works without introducing significant bias. Additionally, to the best of the authors knowledge, no other models are multi-class networks trained to identify not only the fruits but also the flowers and nodes of tomato plants.

For this experiment, three types of backbones were selected: MobileNetV3Small [82], ResNet50V2 [77], and VGG-16 [182]. This selection was driven by the need to compare the original YOLOv5 CSP-Darknet53 backbone with architectures featuring distinct characteristics: a deep stack of convolutional layers (VGG-16), an emphasis on residual layers (ResNet50V2), and an optimized design for small, mobile applications (MobileNetV3Small). The results obtained with these backbones are presented in Table 4.5.

Table 4.5 Results achieved on tomato recognition using different backbones. It can be seen that VGG-16 outperforms the other models in each detection task.

Class	Backbone	TP	M	B-FP	B-FN
Fruit number	MobileNetV3S	64%	1%	7%	35%
	ResNet50V2	73%	3%	7%	24%
	VGG-16	75%	3%	7%	23%
Nodes	MobileNetV3S	50%	0%	48%	50%
	ResNet50V2	64%	0%	54%	36%
	VGG-16	69%	0%	59%	31%
Flowers	MobileNetV3S	48%	1%	45%	51%
	ResNet50V2	56%	1%	39%	44%
	VGG-16	59%	2%	34%	38%



(a) MobileNetV3Small (b) ResNet50V2

(c) VGG-16

Fig. 4.9 F1 score achieved using different backbones.

Interestingly, the three proposed backbones achieve results comparable to YOLOv5s (for MobileNetV3Small), YOLOv5m (for ResNet50V2), and YOLOv5x (for VGG-16). This suggests that the network's effectiveness is more influenced by the total number of parameters rather than the specific types of layers, such as residual layers. Additionally, the F1 scores achieved by the different backbones at varying confidence levels are compared, as shown in Figure 4.9.

The previous figure demonstrates that the network can achieve a peak F1 score of 0.6, which is comparable to those achieved by single models and ensembles. However, this is accomplished with a slightly higher level of confidence in the detection of the bounding boxes.

4.4 Summary

Plant Phenomics is the science of measuring multiple phenotypic traits of plants at once to evaluate their status based on life-cycle conditions. Although deep learning (DL) models are widely acknowledged as the leading computer vision techniques in numerous agricultural fields, they require a significant volume of well-annotated data, which can be expensive and labour-intensive. The swift progress in key enabling technologies has enabled high-throughput phenotyping (HTP), driven by the vast data produced by computer vision systems. In this scenario, artificial intelligence algorithms are essential for this extensive dataset automation, standardization, and quantitative analysis.

Identifying and counting the flowers and fruits on a plant indicates fruit set across different plant varieties or the same variety under varying nutritional treatments. Capturing multiple images during plant growth allows for the assessment of flower development timing, providing insights into plant stress levels. Additionally, evaluating the number and spacing of leaf nodes measures plant development, aiding in the assessment of nutritional product efficacy and the comparison of plant varieties under various stress conditions, particularly abiotic stress. Therefore, the primary objective of this chapter is to focus on identifying flowers, fruits and nodes with a single CNN model. This chapter analyzes a complex image dataset of the aerial parts of tomato plants, meticulously labeled by domain experts who provided bounding boxes for nodes on the main stem, flowers, and fruits. Given the limited research on tomato identification and classification using the YOLOv5 model, this chapter

investigates the application of this architecture for single and multi-class classification of tomato cultivars. Initially, datasets were trained and evaluated using standard YOLOv5 models along with transfer learning techniques. the developments were evaluated using two methods: Test-Time Augmentation (TTA) and model ensembling. These strategies are designed to enhance the performance achieved through transfer learning. Finally, it was evaluated how different backbone architectures affect the detection performance. This entailed testing various network layers and configurations to discern their impact on the accuracy and overall performance of the model.

Chapter 5

Optimizing Tomato Plant Phenotyping Detection: YOLOv8 Model

5.1 Overview

As discussed in the third chapter, section 4.1, The tomato is one of the most extensively studied crops globally, especially in Italy. It is key phenotypic traits, such as flowers and fruits, are vital for reproduction and play a crucial role in the real-time monitoring and evaluation of crop growth. High-throughput phenotyping (HTP), combined with machine learning (ML) and deep learning (DL) techniques, is instrumental in assessing and mitigating issues like shade exposure that can affect fruit quality and differentiation. These methods can effectively distinguish between ripe and unripe fruits. The YOLO (You Only Look Once) model family, known for balancing model complexity with real-time performance, excels at capturing intricate patterns and relationships. Introducing the denser YOLOv5 model allowed further improvement in localization and detection results. Specifically, [24] conducted an investigation on the capabilities of different versions of the YOLOv5 base model, exploring its effectiveness in the identification of three phenotypic traits, that is, flowers, fruits, and nodes, on a challenging dataset. Interestingly, the research demonstrated that the denser YOLOv5 models were able to both reduce false negatives and correctly label objects that were missed on purpose during the labeling step. Other

authors also proposed several enhancements to the original model. For example, [163] modified the standard backbone via Squeeze-and-Excitation (SE) modules, achieving a 94.10% on the proposed dataset. Another proposal was made by [171] with their YOLOv5-4D model, which combined object detection, multiple object tracking, and specific tracking area counting to effectively count tomato clusters, achieving an mAP of 74.8% on the proposed dataset. The authors of [108] modified the YOLOv5s standard model, introducing a stepwise partial network to enhance the inference speed of the network, and replaced the complete loss of Intersection over union (CIoU) with the efficient loss of Intersection over union (EIoU) to optimize the prediction box regression process. These changes improved the mAP of the original YOLOv5s model of 0.66% on the proposed dataset. To further reduce the computational cost associated with the development of denser models, the authors of [230] proposed THYOLO, an algorithm aimed at reducing the computational cost by combining channel pruning and the optimization of key hyper-parameters, achieving an overall reduction of parameters of 84.15%, while keeping comparable performances. Another approach was proposed by SM-YOLOv5, developed by [208], which replaced the original backbone with the MobileNetV3-Large network. This reduced both the computational cost and the model weight, making it deployable easily on constrained robots. The model also achieved interesting results, with an mAP of 98.80% on the proposed dataset. To address the identification of small targets, such as flowers and tomato fruits, [233] proposed a variant to the standard YOLO architecture using a detachable head, thus removing the requirement of pre-determined anchor boxes. The authors used two variants of this network, namely YOLOXMOB and YOLOXPC, which achieved an mAP of 62.10% and 77.33%, respectively, surpassing the value achieved by the bare network on the proposed dataset. Finally, [222] proposed an enhancement to the YOLOv8 architecture specifically tailored for tomato harvesting automation, implementing a feature enhancement module to improve feature extraction, replacing deeply separable convolution with regular convolution to reduce computational complexity, and introducing a two-way attention gate for enhancing the overall recognition accuracy. These modifications lead to an overall mAP of 93.4% on the proposed dataset, reducing the overall number of parameters required.

Researchers have focused on enhancing these model performances through advanced techniques like hyper-parameter optimization. However, this approach faces challenges, particularly with data imbalance—a common real-world issue

where certain phenotypic traits (e.g., nodes) are more densely sampled than others (e.g., fruits and flowers). Additionally, the small size of many traits of interest can lead to suboptimal results with the standard YOLO architecture.

This chapter introduces a framework for phenotypic trait recognition that addresses these challenges using YOLOv8. One key component is the data balancing and augmentation strategy. This strategy improves overall detection performance while ensuring the model's effectiveness and robustness under constrained conditions. It incorporates a data generation phase, utilizing image processing (IP) techniques to artificially create data, and a balancing phase to equalize class distributions. These approaches collectively enhance prediction accuracy. Another crucial aspect is the enhancement of the YOLOv8 architecture. Recognizing that utilizing the basic YOLO architecture may result in suboptimal outcomes due to its insufficient optimization for the task, an attention module is integrated into the YOLOv8 architecture, thereby bolstering its capacity to detect small-sized objects.

Finally, this Chapter does an extensive comparative evaluation with the approach discussed in Chapter Three demonstrating that the proposed method produces superior results, ensuring more reliable detection of tomato plants.

5.2 Methodology of Optimizing Tomato Plant Phenotyping

In this chapter, the goal is to optimize tomato plant phenotyping detection using YOLOv8 to address data complexity. The structure of this chapter includes several key components:

- The first subsection explained data preparation.
- The second subsection details the data balancing method employed.
- The third subsection delves into the Deep Learning algorithms developed, specifically focusing on the YOLOv8 architecture.
- The fourth subsection introduces the integration of the SE-Block Attention Module into the YOLOv8 architecture.

- Finally, the fifth subsection covers the evaluation metrics used to assess the detection and classification capabilities of the proposed methods.

5.2.1 Data Preparation

The dataset utilized in this study comprises 1,673 images, each captured at a standardized resolution of 1624 × 1234 pixels. These images, previously used in the study by [24], were collected at the HTP platform located at the ALSIA Metapontum Agrobios Research Centre. The dataset includes three categories of objects related to phenotypic traits: flowers, fruits, and nodes.

It is important to note that the intrinsic structure of the tomato plant complicates the recognition of these traits. For instance, some branches may bear clusters of large tomatoes, while others may have smaller ones. Additionally, fruits positioned close to nodes can cause overlaps, complicating accurate analysis by the model. These challenges are further intensified by data imbalance in real-world scenarios. The labeled dataset exhibits an imbalanced class distribution, with the node class having more than twice the number of samples compared to the other two classes, as shown in Table 5.1. This imbalance between nodes, flowers, and fruits can impact the model classification accuracy. There is a risk that the model might misclassify nodes as fruits due to their similar green color, particularly when fruits are unripe. To address this issue, balancing the class distribution could help alleviate some of the challenges.

Table 5.1 Number of instances per class before and after data balancing.

Class	Before balancing	After balancing
Fruit	1862	4614
Nodes	9276	4744
Flowers	3111	4925

Moreover, as fruits grow, they transition to a yellow hue. When smaller, they may resemble flowers, which are also yellow, posing another risk of misclassification between flowers and fruits.

5.2.2 Data Balance

The primary challenge discussed in this chapter is the class imbalance within the dataset categories. Imbalanced datasets create significant obstacles for data-driven algorithms due to the unequal distribution of samples across classes. This imbalance can lead to biases and reduce the algorithm ability to learn effectively from under-represented classes [174]. Consequently, the algorithm generalization capability is weakened, which can result in inaccuracies and poor performance in real-world applications.

Various strategies exist to address data imbalances. In this context, a simple random image-based under-sampling approach is not appropriate, as it may target images instead of labels, potentially eliminating minority-class objects rather than majority-class ones. An alternative strategy might involve selectively reducing the number of objects from the majority class. However, this approach carries the risk of increasing false positives during predictions, as the model might find it challenging to accurately learn and identify nodes.

In addressing these obstacles, this chapter introduces a method that strategically merges flowers and fruit with existing nodes, effectively camouflaging the nodes from the model attention. As a result, the model disregards them during prediction. Essentially, this involves amalgamating data to produce additional samples for underrepresented classes. Additionally, new instances were meticulously crafted to closely resemble the characteristics of the minority classes. These were generated through various manipulation techniques applied to existing data.

To implement this strategy, an examination was carried out to identify instances in the majority class that had no overlap with other classes, indicated by an Intersection over Union (IoU) value of 0. The same assessment was conducted for minority classes. This analysis determined the quantity of new samples needed. Subsequently, the identified objects were merged with original images from the dataset to create new instances. The transformation in the number of instances per class before and after data balancing is outlined in Table 5.1.

The following formula can be employed to compute the requisite number of samples for achieving equilibrium:

$$\frac{N_{\text{fruit}} + N_{\text{flower}} + X}{(N_{\text{node}} - X) + (N_{\text{fruit}} + N_{\text{flower}} + X)} = \frac{2}{3} \tag{5.1}$$

Where:

- N_{fruit} represents the number of fruit class.
- N_{flower} represents the number of flower class.
- N_{node} represents the number of node class.
- X represents the number of samples deducted from N_{node} and added to N_{fruit} and N_{flower}.

Hence, the method suggested here produced fresh data by placing fruit and flowers at the coordinates of vacant nodes, i.e., nodes devoid of any existing intersection with fruit and flowers. This methodology corresponds with a similar approach outlined by [174]. Figure 6.1 illustrates the contrast between the original instances before and after implementing data balancing.

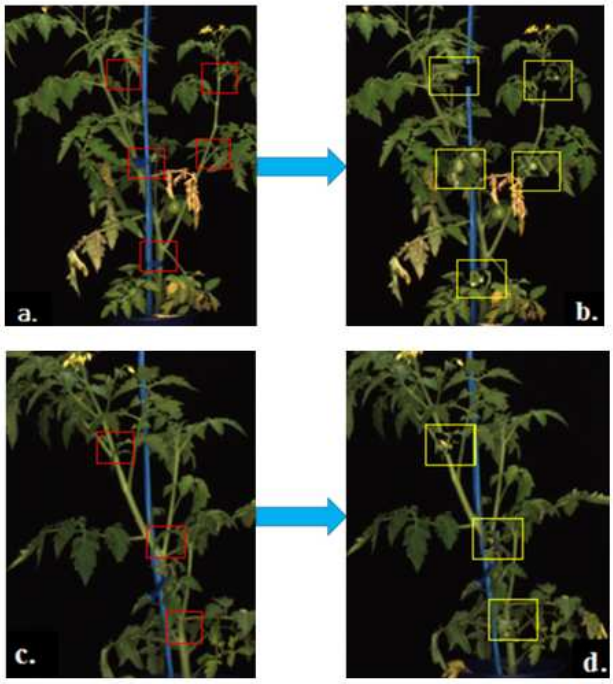


Fig. 5.1 Comparison between an instance of the dataset before and after the data balancing process. (a) Instance with an empty node; (b) Effect of adding fruits on empty nodes; (c)Instance with an empty node; (d) Effect of adding flowers on empty nodes.

5.2.3 YOLOv8 Architecture

The groundbreaking You Only Look Once (YOLO) framework was pioneered by [164], marking a significant advancement in object detection. YOLO introduced an end-to-end network capable of both detecting object bounding boxes and classifying their labels simultaneously. Since its inception, YOLO has undergone several iterations, culminating in its eighth version in January 2023 [88]. The latest iteration prioritizes the following pivotal components:

- **Backbone**: YOLOv8 backbone leverages a modified version of the *Cross Partial Stage* (CSP) technique [204], which divides the feature map into distinct components for convolution operations and their outputs. This division reduces overall computational complexity while preserving the detector learning capability. YOLOv8 backbone builds upon the C2f module, which is a faster adaptation of CSP inspired by the ELAN structure in YOLOv7 [203]. Additionally, integration of the SPPF module enhances detection across varied scales.
- **Neck**: YOLOv8 neck employs the PAN-FPN module to fuse features effectively across diverse scales. This module employs a multi-scale fusion strategy utilizing both FPN and PAN architectures. The upper layers capture richer information, while the lower layers retain precise location details.
- **Head**: YOLOv8 introduces a decoupled head architecture, separating the classification and detection processes. Departing from the previous anchor-based method, YOLOv8 adopts an anchor-free approach, pinpointing object locations based on their centers and predicting distances to the bounding box, thereby eliminating the need for predefined anchors.

A visual representation illustrating the structure of the YOLOv8 model is depicted in Figure 5.2.

The decision to utilize YOLOv8 as the foundational model in this chapter stemmed from its lightweight architecture, allowing for real-time object detection, and its versatility across various scales. Figure 5.3 depicts the overarching framework employed in this study.

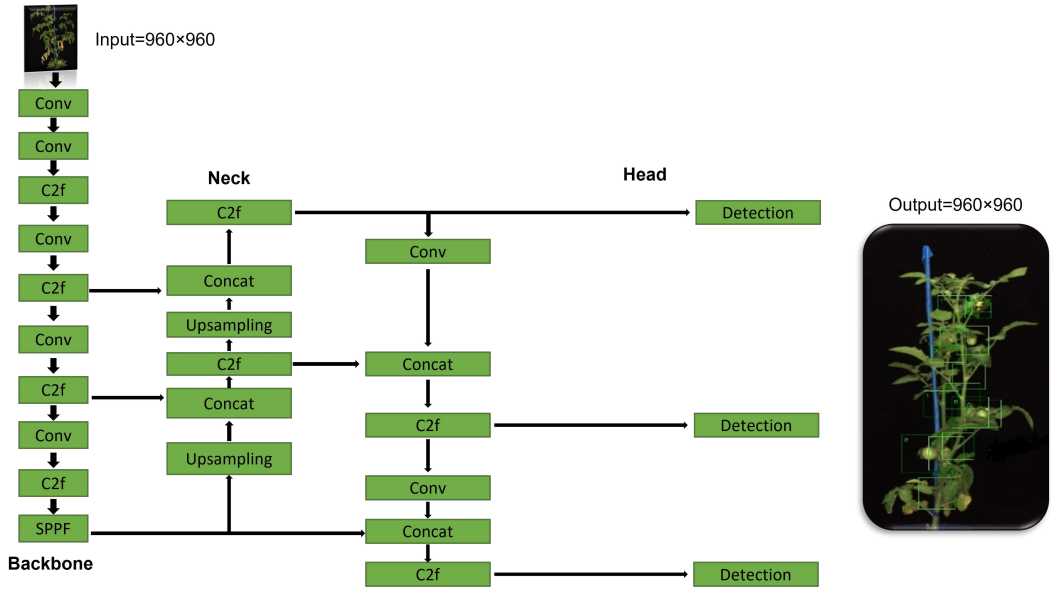


Fig. 5.2 The architecture of the YOLOv8 model.

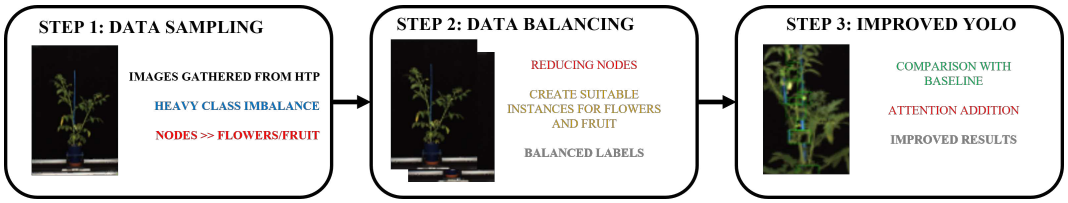


Fig. 5.3 The processing framework proposed within this work. First, images are gathered directly from a data source like an HTP platform. Then, data augmentation and balancing steps are used to gather a suitable dataset. Finally, several improvements are added to the bare YOLOv8 architecture to improve results.

5.2.4 SE-Block Attention Module

The Squeeze-and-Excitation (SE) block, introduced by [83], is a popular attention mechanism designed to capture the interdependencies between channels. This technique allows the network to adjust its features dynamically, enhancing important features by leveraging global information and reducing the emphasis on less significant ones [122].

A significant challenge in tomato image analysis is the color similarity among nodes, unripe fruits, and the background, typically including leaves. Additionally, the small size of the target classes (flowers, fruits, and nodes) makes accurate

differentiation more difficult. As a result, the model might inadvertently allocate its weights uniformly across the entire image dataset. Thus, a strategic approach is necessary since many of these images are not useful for our purposes. To address this issue, the SE module was integrated as a preprocessing step in the head of the YOLOv8 model to refocus the model's attention on the relevant classes.

The placement of the SE-block module is illustrated in Figure 5.4, was determined based on the recommendations of the original authors. They noted that while the ideal placement may require empirical determination, the SE-block typically performs suboptimally when positioned after a concat operation [83].

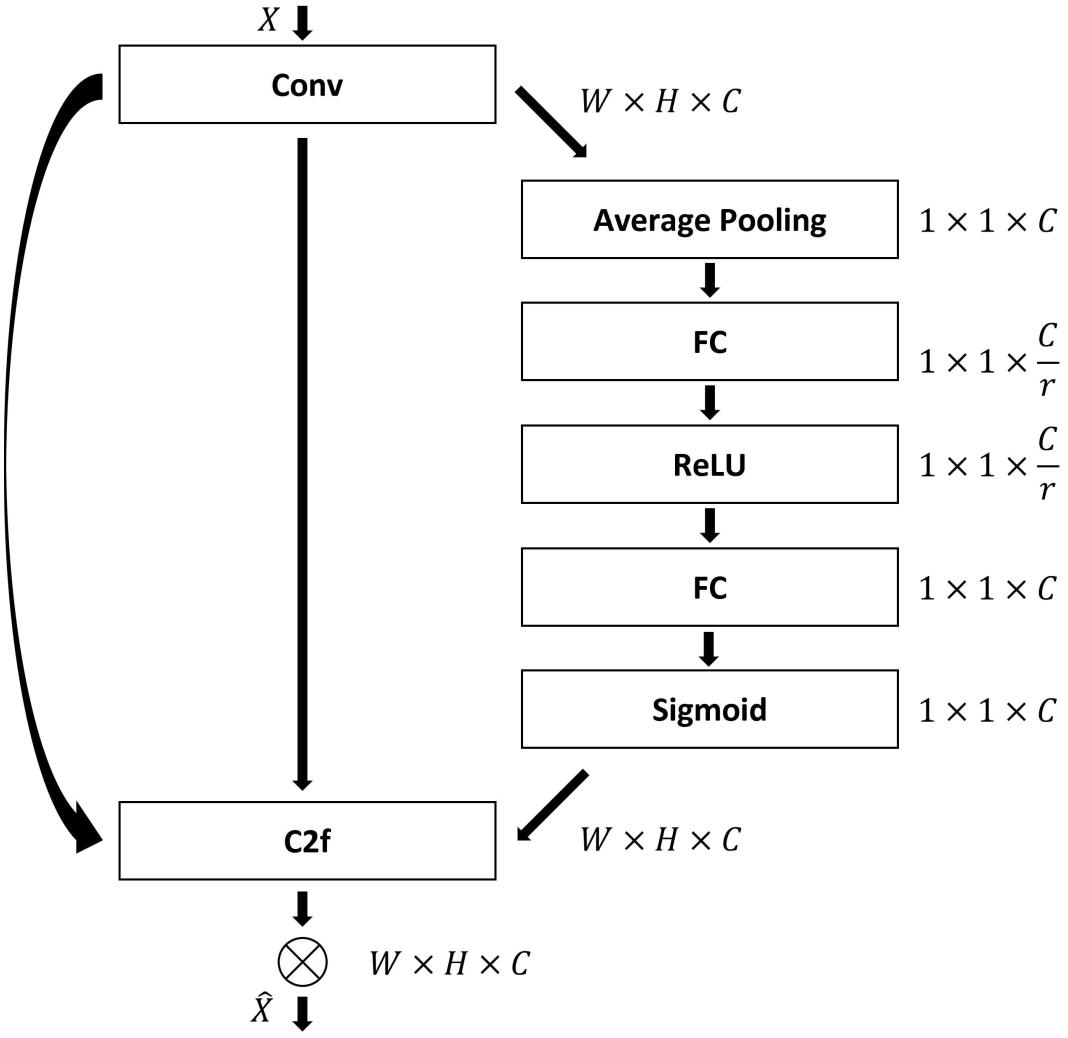


Fig. 5.4 The result of embedding the SE-block within the C2f and Conv modules.

Therefore, the SE-block module was integrated after the C2f modules of the original architecture to enhance the focus on the features extracted from these layers. The modified architecture of YOLOv8 with the SE-block module is depicted in Figure 5.5.

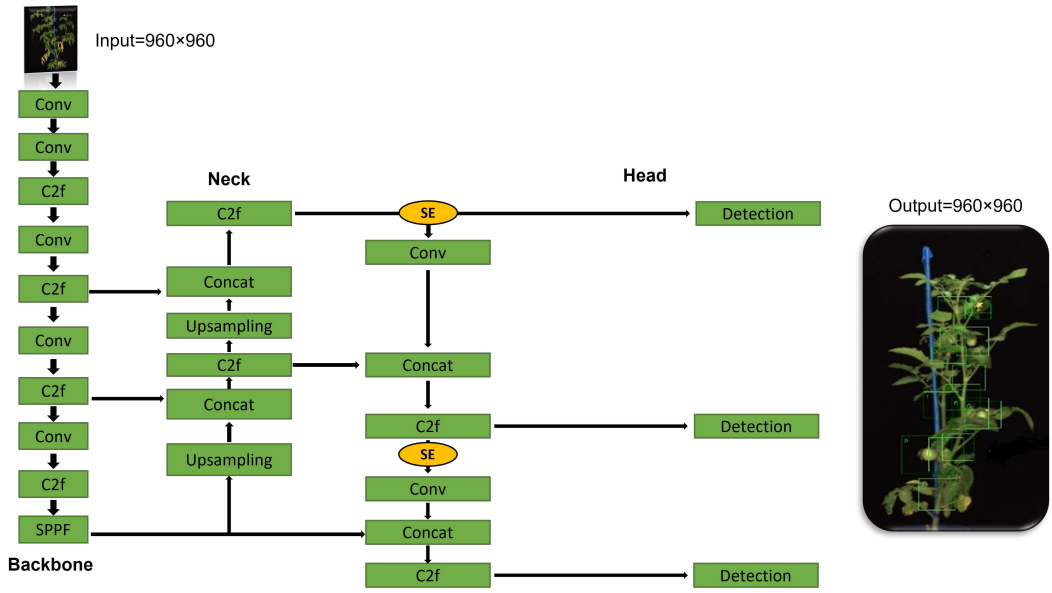


Fig. 5.5 The proposed architecture, with the addition of the SE-block modules.

A SE-block constitutes a computational unit that can be constructed using a transformation F_{tr}, which maps an input $X \in \mathbb{R}^{H' \times W' \times C'}$ to feature maps $U \in \mathbb{R}^{H \times W \times C}$.

Therefore, the output can be written as:

$$y_c = v_c \cdot X = \sum_{s=1}^{C'} v_c^s \cdot x^s \tag{5.2}$$

Addressing channel dependencies, the focus is on channel-specific signals in output features. However, local receptive fields limit contextual exploitation. Global spatial data were incorporated into channel descriptors using global average pooling to overcome this, producing statistics $Z \in \mathbb{R}^C$.

$$z_c = F_{\text{sq}}(y_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} y_c(i, j) \tag{5.3}$$

Channel dependencies are consequently considered to harness squeezed information. This operation must be flexible, accommodating nonlinear interactions and supporting non-mutually-exclusive relationships. This was achieved using a sigmoid-activated gating mechanism to meet these criteria.

$$s = F_{\text{ex}}(z, W) = \sigma(g(z, W)) = \sigma(W_2 \delta(W_1 z)) \tag{5.4}$$

For model simplicity and enhanced generalization, the SE-block used in this work contains two fully connected layers around the non-linearity.

$$\tilde{x}_c = F_{\text{scale}}(y_c, s_c) = s_c \cdot y_c \tag{5.5}$$

5.3 Experimental Results and Discussion

The study introduces and implements the YOLOv8 architecture, enhancing its performance in tomato detection through data balancing and the integration of the SE-Block attention module. By balancing a dataset of 1683 tomato images, significant improvements are observed compared to the results in Chapter 4.

This section is organized into seven parts:

- The first part explains the experimental setup.
- The second part compares single-stage detectors (YOLOv8) with two-stage detectors (R-CNN).
- The third part explains the data augmentation techniques used.
- The fourth part compares the SGD and Adam optimizers to determine the best choice.
- The fifth part explores how data balancing impacts model performance.
- The sixth part analyzes the effect of adding the SE-Block attention module to the model.
- The seventh part compares YOLOv8 with YOLOv5 as discussed in Chapter 4.

5.3.1 Experimental setup

YOLOv8, by default, processes images with a fixed size of 640 × 640 pixels. This resolution helps manage the high memory and computational demands of the network, making it feasible to train denser models on relatively limited hardware. However, this resolution may compromise the visual details of objects of interest, especially if they occupy only small portions of the original image. To balance detail capture and computational feasibility, the images were resized to 960 × 960 pixels before being fed into the network. In the training phase, two optimization algorithms were explored: stochastic gradient descent (SGD) and Adam. This comparison was motivated by a plethora of research findings. Notably, [226] highlight SGD speed and low computational overhead, tempered by its vulnerability to fixed learning rates, as echoed by [46], [124], and [215]. To tackle this limitation, [26] propose employing a dynamic learning rate that gradually diminishes during training. Conversely, Adam exhibits diminished sensitivity to learning rates, supported by research from [116], and yields more precise gradient estimates, as underscored by [97]. However, it is essential to note that, as cautioned by [213], Adam generalization capabilities may lag behind those of SGD.

A standardized set of parameters was meticulously chosen to ensure equitable comparisons among the algorithms, detailed in Table 5.2. Notably, the adaptive learning rate played a crucial role, gradually diminishing from 0.01 to 0.001 throughout the training process. This adjustment ensured consistent and fair evaluations across the board.

Table 5.2 Training parameters settings

Parameter	Values
Batch size	4
Image size	960×960
Initial learning rate	0.01
Final learning rate	0.001
Weight decay	0.937
Momentum	0.0005

5.3.2 Comparison with Two-Stages Detectors

To validate the effectiveness of the proposed approach, a comparison was conducted with the state-of-the-art Faster R-CNN method. Introduced by [66], Faster R-CNN operates as a two-stage detector. The first stage involves a region proposal network that identifies candidate regions in the image for object localization. The second stage involves a classification model that assigns the most probable class to the objects within these candidate regions.

The baseline YOLOv8n model was compared with Fast R-CNN to ensure a fair evaluation. Both models were trained on the imbalanced dataset for 100 epochs. This comparison aimed to assess whether the bare YOLOv8 model possesses greater representational capability than one of the top-performing two-stage detectors. The results, presented in terms of mAP, are shown in Table 5.3.

Table 5.3 Results of the comparison between YOLOv8n and Fast R-CNN on imbalanced data

Model	mAP0.5	mAP0.95
YOLOv8n	**65.08%**	**19.12%**
Fast R-CNN	26.29%	7.50%

The results demonstrate that the baseline model significantly outperforms Fast R-CNN on the proposed dataset. This is further corroborated by the predictions made by Fast R-CNN, as illustrated in Figure 5.6. While the network was able to partially identify objects belonging to the majority class (i.e., nodes), it struggled to accurately identify traits associated with the two minority classes. Therefore, it is reasonable to conclude that YOLOv8 is superior to Fast R-CNN in this specific context and should be chosen as the base architecture for object identification.

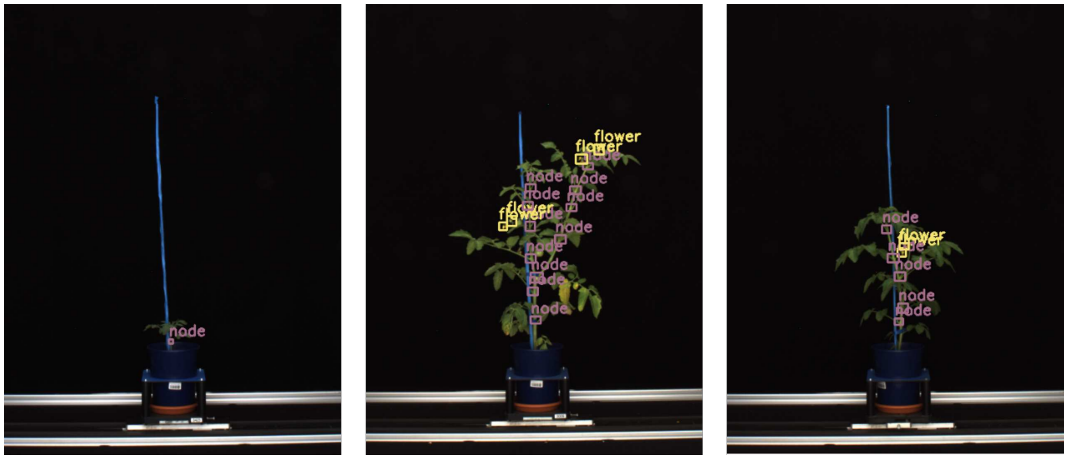


Fig. 5.6 Predictions performed by Fast R-CNN

5.3.3 Data Augmentation

The effectiveness of data augmentation was evaluated using the base YOLOv8n model. Specifically, four transformations were tested:

- HSV: New images were generated with an increment in the V value to enhance the contrast between fruits and nodes.
- Translate: New images were created by translating different patches of the original images.
- Scale: New images were produced by scaling the original ones.
- Flip: New instances were generated by randomly flipping the original images either vertically or horizontally.

Table 5.4 Comparison of the results provided by using different data augmentation methods on YOLOv8n with the proposed approach

Hyper-parameter	Class	mAP50%	mAP50 – 95%
Baseline	Fruit	66.20%	19.26%
	Flower	57.31%	18.34%
	Node	**65.08%**	19.12%
HSV (V)	Fruit	**68.30%**	19.44%
	Flower	59.07%	18.34%
	Node	63.03%	19.02%
Translate	Fruit	67.25%	**20.49%**
	Flower	**64.35%**	19.17%
	Node	57.02%	**20.11%**
Flip	Fruit	66.37%	20.14%
	Flower	61.24%	19.27%
	Node	64.02%	20.01%
Scale	Fruit	**68.30%**	20.14%
	Flower	61.14%	**20.10%**
	Node	62.07%	20.04%

The results presented in Table 5.4 indicate modest improvements in mAP scores, particularly for minority classes, with the application of HSV, Translate, and Scale transformations. Notably, the Scale transformation had the most significant impact, yielding the highest mAP performance across minority classes. Therefore, it is reasonable to conclude that data augmentation enhances the model sensitivity, enabling better detection of small objects, which is especially advantageous given the unique characteristics of the dataset utilized in this study.

5.3.4 Comparison of the SGD and adam optimizers

In this section, the performance of both optimizers is compared. The comparison utilizes the full range of densities provided by YOLOv8, from the sparser YOLOv8n model to the denser YOLOv8x model. It is important to note that this evaluation was conducted on the base dataset, without any augmentation, to ensure a fair assessment of the algorithm effectiveness. The results are presented in Table 5.5.

Table 5.5 Results of evaluating different data augmentation methods on YOLOv8n with the proposed approach.

Optimizer	Models	P%	R%	F1	mAP50%	mAP50-95%
SGD	YOLOv8n	58.63%	58.98%	58.80%	53.69%	18.75%
	YOLOv8s	62.56%	62.90%	62.72%	58.25%	20.56%
	YOLOv8m	63.93%	61.35%	62.61%	58.14%	20.53%
	YOLOv8l	62.49%	62.26%	62.37%	56.88%	20.22%
	YOLOv8x	**69.78%**	**63.12%**	**65.28%**	**64.09%**	**23.15%**
Adam	YOLOv8n	57.08%	57.12%	57.09%	52.70%	18.10%
	YOLOv8s	61.36%	59.61%	60.47%	55.31%	19.63%
	YOLOv8m	60.43%	61.19%	60.80%	56.19%	19.70%
	YOLOv8l	61.16%	60.44%	60.79%	55.07%	18.68%
	YOLOv8x	58.79%	59.66%	59.22%	53.98%	18.86%

The results in Table 5.5 demonstrate that SGD outperforms Adam across all metrics. Notably, the best-performing model, YOLOv8x, significantly improves precision (approximately 10%) and mAP50 (approximately 11%). These findings suggest that SGD is more effective than Adam in this context. This can be attributed to the nature of the dataset, which is imbalanced with a significantly smaller number of samples in the minority classes than in the majority class. The dynamic adjustment of the learning rate in SGD likely played a crucial role in effectively guiding the optimization process under these conditions. In contrast, the Adam adaptive learning rate algorithm may have been less suited to this scenario. Additionally, the simplicity of the SGD update rule likely helped prevent overfitting the limited data available for the minority classes. By incorporating class weights during training, SGD implemented a prioritized learning approach, assigning higher weights to instances from minority classes, thus resulting in observed performance improvements.

Consequently, the decision was made to proceed with the remaining tests using the SGD optimizer. However, an important aspect that still needs to be addressed is the impact of balancing the data distribution on the analysis outcomes.

5.3.5 Data Balancing

The impact of data balancing was evaluated by comparing the performance of different models on imbalanced versus balanced data. The results are shown in Table 5.6.

The results in Table 5.6 demonstrate the impact of data balancing, with improvements ranging from 3% to 8% across all metrics for nearly all the proposed models.

Let us briefly consider the results achieved. As demonstrated in Section 5.3.4, YOLOv8x achieves the best results, likely due to its large model capacity. However, these results are biased towards the majority class, suggesting that the model may have reduced generalization capabilities. When the data are balanced, smaller models achieve performance comparable to YOLOv8x, primarily because the adequate amount of samples properly characterizes the data generation mechanism. Consequently, with balanced data, YOLOv8l achieves the best performance. Interestingly, YOLOv8x shows a performance decrease with balanced data, likely due to the double descent phenomenon [147].

Table 5.6 Results of the comparison between imbalanced and balanced data

Dataset	Models	P%	R%	F1	mAP50%	mAP50-95%
Imbalance	YOLOv8n	58.63%	58.98%	58.80%	53.69%	18.75%
	YOLOv8s	62.56%	62.90%	62.72%	58.25%	20.56%
	YOLOv8m	63.93%	61.35%	62.61%	58.14%	20.53%
	YOLOv8l	62.49%	62.26%	62.37%	56.88%	20.22%
	YOLOv8x	69.78%	**63.12%**	**66.28%**	**64.09%**	**23.15%**
Balance	YOLOv8n	64.65%	58.35%	61.33%	59.77%	21.42%
	YOLOv8s	66.81%	60.55%	63.52%	61.40%	22.22%
	YOLOv8m	69.32%	60.31%	64.50%	61.94%	22.69%
	YOLOv8l	**70.84%**	60.73%	65.39%	62.11%	22.44%
	YOLOv8x	69.37%	59.52%	64.06%	61.66%	22.12%

5.3.6 SE-block Attention Module

After applying data balancing, the effects of embedding the SE-block attention module were evaluated, as described in Section 5.2.4. The results are presented in Table 5.7.

Table 5.7 Results of evaluating balanced data using the attention mechanism

Model	P%	R%	F1	mAP50%	mAP50-95%
YOLOv8n	66.08%	52.23%	58.34%	55.85%	18.77%
YOLOv8s	66.37%	58.40%	62.13%	60.47%	21.23%
YOLOv8m	68.49%	58.74%	63.24%	60.03%	20.49%
YOLOv8l	68.52%	**60.68%**	**64.36%**	60.52%	21.33%
YOLOv8x	**69.23%**	56.87%	62.44%	**60.57%**	**21.94%**

Interestingly, when the attention module is applied to the balanced dataset, there is a performance decrease across all versions of YOLOv8. This can be attributed to the data balancing process itself, which can lead to information loss regardless of the augmentation technique used. Specifically, data balancing may involve duplicating existing objects multiple times, potentially distorting the underlying data generation mechanisms. Since attention mechanisms focus on local information, which may be biased by this augmentation, performance can suffer. Additionally, as discussed in Section 5.3.5, the reduced information about the nodes can further impact the overall effectiveness of the model.

Therefore, the performance of the modified version of YOLOv8 should also be evaluated on imbalanced data. The results of this evaluation are shown in Table 5.8.

Table 5.8 Results of evaluating imbalanced data using the attention mechanism

Model	*P%*	*R%*	F1	mAP50%	mAP50-95%
YOLOv8n	69.48%	57.31%	62.81%	60.85%	20.90%
YOLOv8s	68.66%	62.51%	65.44%	64.25%	22.98%
YOLOv8m	**71.03%**	**63.66%**	**67.14%**	**65.82%**	**23.70%**
YOLOv8l	70.50%	63.31%	66.71%	64.62%	22.61%
YOLOv8x	69.60%	64.01%	66.68%	64.67%	22.99%

It becomes evident that the models yield improved results on imbalanced data when the attention module is used. Again, this is because imbalanced data retains a broader spectrum of available information, which, in this particular situation, may be able to provide better results if compared with data balanced with the method previously proposed.

To address the challenge of information loss while still leveraging data balancing, an alternative approach was implemented. Instead of using balanced data directly, the model was trained on the original imbalanced data, assigning higher weights to instances from minority classes. This method balanced the data classes while preserving the original information. A "transfer learning-like" strategy was used by applying the weights from the model trained on balanced data to the imbalanced data. This approach led to a noticeable improvement in the results, even when compared to those achieved on the imbalanced data alone, as shown in Table 5.9.

Table 5.9 Results of evaluating imbalanced data using the attention mechanism and pre-trained weights obtained from the balanced dataset

Models	P%	R%	F1	mAP50%	mAP50-95%
YOLOv8n	70.01%	60.69%	65.01%	63.69%	22.33%
YOLOv8s	70.20%	64.61%	67.28%	65.65%	23.27%
YOLOv8m	**71.59%**	**64.96%**	**68.11%**	**65.77%**	**23.38%**
YOLOv8l	70.11%	65.59%	67.77%	64.82%	22.61%
YOLOv8x	71.17%	62.15%	66.35%	64.83%	23.89%

5.3.7 YOLOv8 vs YOLOv5

The final comparison outlined in this paper aims to evaluate the outcomes of this study about the methodology presented by [24]. It is worth noting that [24] previously examined the utilization of YOLOv5 on the identical dataset employed in this study. However, it is crucial to emphasize that their evaluation focused solely on imbalanced data, without incorporating attention mechanisms into the model architecture. Consequently, this comparative analysis serves to gauge the impact of the proposed balancing and attention mechanisms. Specifically, the performance of the denser architectures, namely YOLOv5x and YOLOv8x, was exclusively assessed.

The disparities between the two methodologies were scrutinized in terms of precision, recall, and F1 score.

Precision will be examined as shown in Figure 5.7. Both networks demonstrate similar outcomes, although YOLOv5x exhibits a notable decrease in precision around a confidence threshold of 0.7, whereas YOLOv8x consistently outperforms when balancing and attention mechanisms are applied. Notably, there is a further decline in precision at a confidence score of 0.8, particularly evident in the node class. This trend might signify an inherent constraint within the model, warranting attention in future research endeavors.

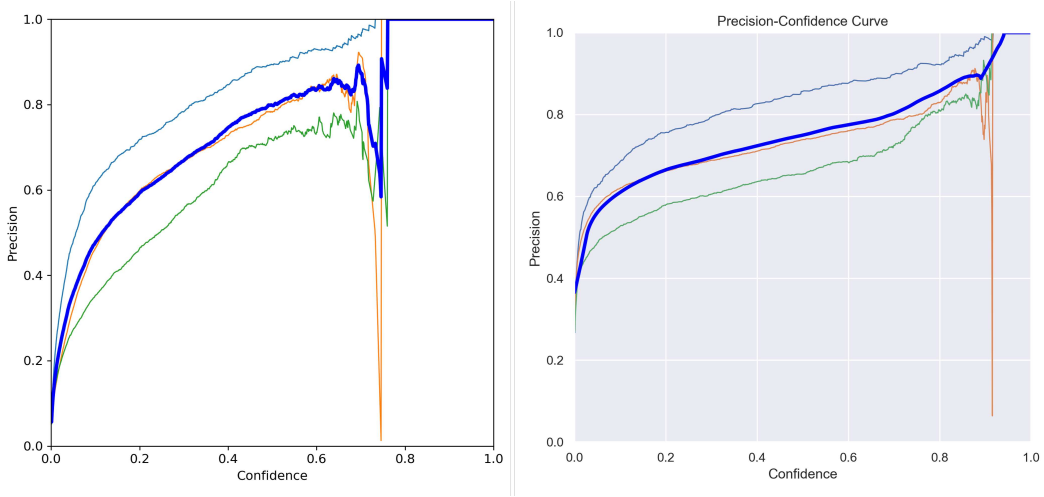


Fig. 5.7 The precision achieved by the YOLOv5x model (on the left) and the YOLOv8x model (on the right) after applying data balancing and attention. Light blue results are for fruit, orange for nodes, and green for flowers.

The findings are further supported by the recall data, depicted in Figure 5.8, where YOLOv8x consistently demonstrates superior performance compared to YOLOv5x.

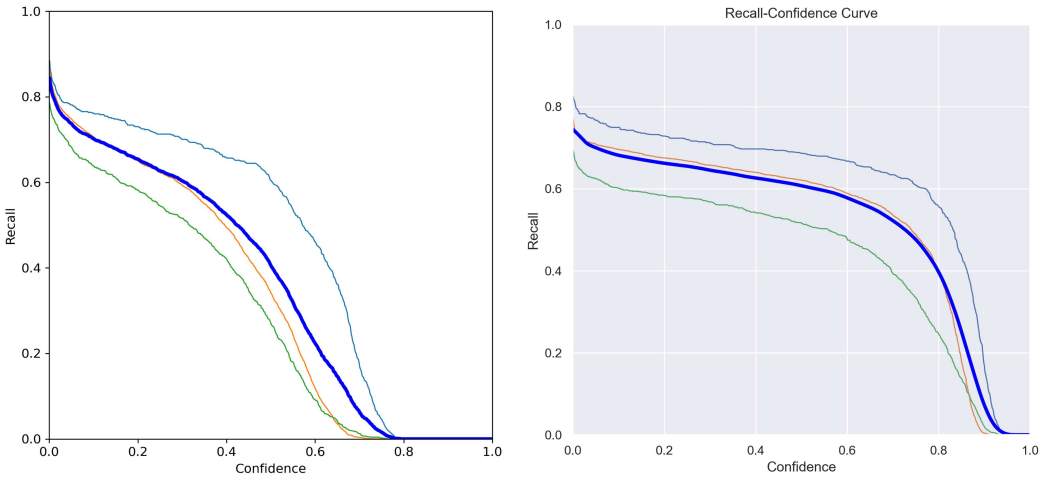


Fig. 5.8 The recall achieved by the YOLOv5x model (on the left) and the YOLOv8x model (on the right) after applying data balancing and attention. Light blue results are for fruit, orange for nodes, and green for flowers.

Let us examine the F1 scores as well, illustrated in Figure. 5.9. For YOLOv5, the highest F1 score was attained at a confidence threshold of 0.4. Conversely, YOLOv8 demonstrated a steady F1 score, reaching it is peak around a confidence threshold of approximately 0.6.

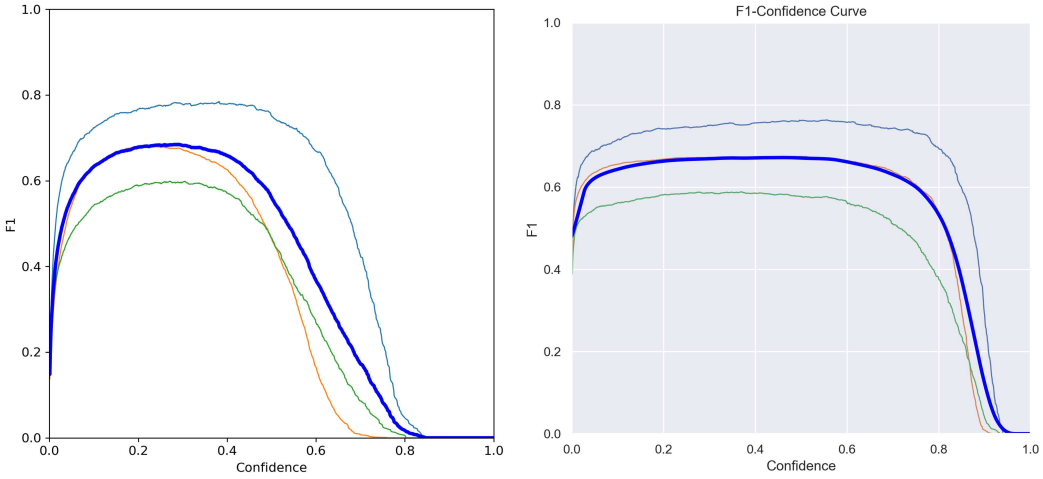


Fig. 5.9 The F1-score achieved by the YOLOv5x model (on the left) and the YOLOv8x model (on the right) after applying data balancing and attention. Light blue results are for fruit, orange for nodes, and green for flowers.

This analysis reaffirms that YOLOv8 excels in accurately identifying objects of interest with increased confidence. This improvement can be attributed to enhancements in the network architecture, resulting in enhanced representation capabilities, as well as the integration of attention mechanisms, enabling precise characterization of small regions of interest.

Lastly, let us explore the outcomes obtained by both methodologies, as detailed in Table 5.10. In this context, B-FP denotes background false positives, signifying boxes identified by the model without corresponding labels validated by domain experts. Conversely, B-FN indicates background false negatives, indicating labeled bounding boxes overlooked by the network.

While the incidence of B-FNs in the YOLOv8 model was comparatively lower than in YOLOv5, both instances of nodes and flowers still exhibit a notable level of B-FN. This suggests that while data balancing in YOLOv8 has somewhat alleviated this concern, as observed in prior research, the fundamental issue likely lies in the model struggle to accurately characterize the visual attributes of these object classes. This challenge also extends to B-FPs, highlighting the model's difficulty in distinguishing between primary and secondary stem nodes, or even between nodes and visually overlapping leaves. Nonetheless, the comparison underscores YOLOv8 consistent outperformance of YOLOv5 across all provided metrics, regardless of the density considered, owing to the introduced innovations.

Table 5.10 Comparison between the results achieved by YOLOv8 and YOLOv5 in [24]

Class	Density	YOLOv8			YOLOv5		
		TP	B-FN	B-FP	TP	B-FN	B-FP
Fruit	Small	75.97%	13.75%	20.89%	64.55%	35.71%	70.69%
	Medium	**79.24%**	**11.75%**	18.36%	73.36%	24.57%	70.29%
	Large	79.04%	12.28%	**18.22%**	76.17%	22.30%	70.69%
	eXtra	78.77%	16.36%	19.49%	77.97%	19.83%	80.84%
Nodes	Small	56.26%	47.69%	44.56%	50.45%	48.99%	48.40%
	Medium	61.49%	52.16%	**39.14%**	64.84%	54.00%	54.89%
	Large	60.05%	57.84%	40.66%	**66.66%**	58.83%	58.55%
	eXtra	55.04%	**46.92%**	45.67%	68.71%	59.64%	59.25%
Flowers	Small	**68.02%**	40.02%	**31.86%**	48.41%	51.59%	45.84%
	Medium	67.38%	37.28%	31.90%	56.81%	44.07%	39.49%
	Large	67.34%	**31.50%**	32.02%	64.12%	34.35%	34.71%
	eXtra	65.09%	38.85%	33.67%	64.52%	34.39%	33.15%

The table 5.10 provides a detailed evaluation of YOLOv8 and YOLOv5 models in detecting objects from three distinct classes—Fruit, Nodes, and Flowers—across varying bounding box densities: Small, Medium, Large, and eXtra. The metrics assessed include True Positives (TP), Bounding Box False Negatives (B-FN), and Bounding Box False Positives (B-FP), offering insights into each model's detection capability and robustness in relation to object size and density.

- Effect of Bounding Box Size on True Positives (TP): Bounding box size plays a significant role in detection performance. YOLOv8 consistently demonstrates higher TP rates across all densities and classes compared to YOLOv5. For medium-sized bounding boxes, YOLOv8 achieves its highest TP rate for the Fruit class (79.24%), outperforming YOLOv5 (73.36%). This indicates that YOLOv8 has a superior ability to accurately detect objects of moderate size, likely due to improved localization and feature extraction capabilities. For small bounding boxes, YOLOv8 detects objects more effectively than YOLOv5, particularly in Flowers (68.02% vs. 48.41%). Smaller bounding boxes typically pose challenges for object detectors due to less spatial information; YOLOv8's better performance here suggests enhancements in detecting fine details.

- Bounding Box False Negatives (B-FN): Missed Detections Bounding box size influences the likelihood of missed detections. Medium-sized objects show the greatest improvement with YOLOv8, where the B-FN is significantly lower across all classes compared to YOLOv5. For example, in the Fruit class, YOLOv8 achieves an 11.75% B-FN compared to YOLOv5's 24.57%, suggesting that YOLOv8 has a higher recall for objects in this density range. However, for smaller bounding boxes (e.g., Nodes and Flowers classes), B-FN rates remain relatively high for both models, with YOLOv5 performing slightly worse. This reflects the challenge of detecting objects with very small bounding boxes, where feature overlap with the background can result in missed detections.

- Bounding Box False Positives (B-FP): Incorrect Detections Bounding box size also correlates with the occurrence of false positives: YOLOv8 demonstrates fewer B-FPs compared to YOLOv5 for medium and large bounding boxes, particularly in the Fruit and Nodes classes. For instance, in the Fruit class with medium-sized bounding boxes, YOLOv8 records a B-FP of 18.36% compared to YOLOv5's 70.29%. This significant difference highlights YOLOv8's precision in avoiding redundant or incorrect predictions. For small bounding boxes, B-FP rates are generally higher in both models. This is likely due to the challenge of differentiating smaller objects from background noise or overlapping bounding boxes, where inaccuracies in localization can lead to spurious detections.

- General Trends Across Classes: Fruit Class: YOLOv8 shows a marked advantage in TP and B-FN rates across all bounding box sizes. Its lower B-FP rates for medium and large bounding boxes further underline its effectiveness in accurately identifying objects with moderate-to-large sizes. Nodes Class: While both models struggle with small bounding boxes (high B-FN and B-FP rates), YOLOv8 outperforms YOLOv5 for medium and large sizes, achieving higher TP and lower B-FP values. Flowers Class: Small bounding boxes in the Flowers class present a challenge for both models; however, YOLOv8 achieves significantly higher TPs (68.02% vs. 48.41%) and slightly lower B-FPs, indicating better precision and recall.

5.4 Summary

Efficient identification and monitoring of tomato plant traits play a critical role in optimizing growth and harvest evaluations. However, conducting stress experiments across diverse tomato genotypes presents challenges due to skewed data distributions. This imbalance can result in classification errors, thereby impeding accurate recognition of essential plant features like flowers, fruits, and nodes.

To address these challenges, this chapter emphasizes a robust strategy of data balancing and augmentation to enhance overall detection performance while maintaining model effectiveness and deployability under constrained conditions. This strategy involves generating synthetic data through relevant image processing techniques and balancing classes to improve prediction accuracy. Additionally, the YOLOv8 architecture is enhanced with an attention module, which boosts the network efficiency in detecting small-sized objects within complex environments.

The approach capitalizes on the capabilities of the YOLOv8 deep learning model, known for its proficiency in handling varied object sizes. Experimental results underscore significant enhancements in model precision achieved through the implementation of data balancing techniques. Specifically, pre-training with optimized weights from balanced data and integrating the SE-block module contribute to notable improvements in detection outcomes. This method not only enhances accuracy but also facilitates the identification of crucial phenotypical traits even in challenging scenarios.

While the study primarily focuses on tomato plant datasets, its methodology can be seamlessly adapted to similar contexts involving other plant species or domains with minimal modifications. Future research aims to further refine this approach and explore its applicability across diverse agricultural scenarios, ensuring robust performance and scalability in trait detection and monitoring systems.

Chapter 6

Enhancing Small Object Detection in the YOLOv8 Model

6.1 Overview

Strawberry is another renowned Italian crop, where precise identification of plant characteristics, particularly fruits and flowers, is essential for precision farming. Accurate identification helps optimize cultivation practices, improve yield, and ensure the health of the crop. Strawberry fruits, despite their small size, possess an innate ability to blend seamlessly with the surrounding foliage. This camouflage effect, coupled with the size disparity between the fruit and the background, presents substantial challenges in event detection for deep neural network (DNN) models. These models, which rely on distinguishing features to identify objects, often struggle to accurately detect strawberries amidst the dense and similarly colored leaves.

In recent years, the authors of [39] improved strawberry maturity detection by enhancing YOLOv4 with MobileNetv3 and depth-wise separable convolution. They reported a detection precision of 98.72% for mature strawberries and 90.76% for immature ones, with average precision (AP) values of 99.56% and 94.00%, respectively. But still multi-stage fruit detection is associated with challenges. In this regard, the introduction of the DSE-YOLO model was able to extract complex details and semantic features horizontally and vertically and increase the accuracy of detecting small fruits by improving the mAP value of 86.58% [209]. However, [85] reported a different claim that the YOLOv5s-MBLS algorithm performs better than the original

YOLOv5 algorithm in response to the challenges faced by existing complex strawberry planting scenarios with a 1.6% increase in the mean average precision and a 21.9% reduction in the model size. [33] by adding ghost convolution (GhostConv), a rotation operator, a block convolution attention module (CBAM), and a light sampling operator called Content-Aware ReAssembly of Features (CARAFE) introduced an advanced YOLOv5 model, which with mAP50% of 94.7 became a promising solution for detecting and provides real-time strawberry disease control. In 2023, [79] introduced a YOLOv5-based custom object detection model, YOLOv5s-Straw, for accurate outdoor strawberry detection. The modification of YOLOv5s involved replacing the C3 module (Concentrated-Comprehensive Convolution) with the C2f module (faster implementation of CSP Bottleneck with 2 convolutions), which improved feature flow. They claimed that under identical conditions, YOLOv5s-Straw was able to achieve the highest mAP at 80.3%, outperforming the other models (YOLOv3-tiny, YOLOv5s, YOLOv5s-C2f, and YOLOv8s). To elevate strawberry recognition during harvest with more advanced models, [27] introduced a real-time and precise detection approach based on the YOLOv7 model. This model had an mAP of 89% and an F1 score of 92%. However, the DSW-YOLO model, as an extension of the YOLOv7 model developed by [49], results marked a 5.0% improvement in precision, 1.7% in the recall, and a 2.2% increase in mAP compared with YOLOv7.

Improving the performance of DNNs in such scenarios requires advanced techniques in model modification. This includes the use of enhanced algorithms and reducing model dimensions to improve resource efficiency, simplify software deployment, and reduce computational costs. Additionally, merging multiple models and designing high-performance, lightweight models can help overcome these challenges, providing more accurate diagnostics and ultimately leading to more effective precision farming practices. With this in mind, this chapter focuses on exploring the latest model in the YOLO family, specifically YOLOv8, to address the complexities of a dataset gathered via a high-throughput phenotyping platform. This study proposes a comprehensive analysis of optimized model head adaptations to enhance the detection of small objects based on the YOLOv8 architecture. These optimizations are pursued while retaining the model's effectiveness in identifying larger fruits and flowers. This approach is expected to yield positive effects and contribute to more accurate and robust detection results on the proposed dataset.

This chapter enhanced the YOLOv8 lightweight network architecture by incorporating a P2 layer and removing the P5 layer to improve small object detection. Additionally, a SE-Block attention module was integrated into the model head section on the P2 layer to amplify the significance of critical information about the classes (fruits and flowers). The SO-YOLOv5 model was created by replacing its head with that of the YOLOv8 model, aiming to reduce model complexity and eliminate the need for manual specification of anchor boxes. Ultimately, the performance of the YOLOv8 baseline model was evaluated against each modified approach (P2-YOLOv8 and SO-YOLOv5).

6.2 Methodology of Enhancing Strawberry Detection

Detecting strawberry plant traits using single-stage detectors (YOLOv8 model) in this chapter is composed of several components. First, the process of image data acquisition and annotation will be reviewed. Next, the YOLOv8 model will be examined as a single-stage detector. Following this, the implementation of the YOLOv8 model will be explained, including the addition of the small object detection head from the P2 layer and the integration of a SE-Block attention module into the model head at the P2 layer. Finally, the implementation of the SO-YOLOv5 model for small object detection will be described.

6.2.1 Image Data Acquisition and Annotation

The chapter utilized data collected from the high throughput plant phenomics platform (HTP) at the ALSIA Metapontum Agro-bios Research Centre in Italy. The dataset was gathered during a stress experiment involving five cultivars of strawberry plants: "Dina", "Candonga", "Limval", "AN15", and "Nsg45". The images with poor quality and no flower and strawberry fruit were removed uniformly by the Python code, and the selected dataset contains 1880 images in PNG format, each with a resolution of 1624 × 1234 pixels. Figure 6.1 illustrates samples of these images.

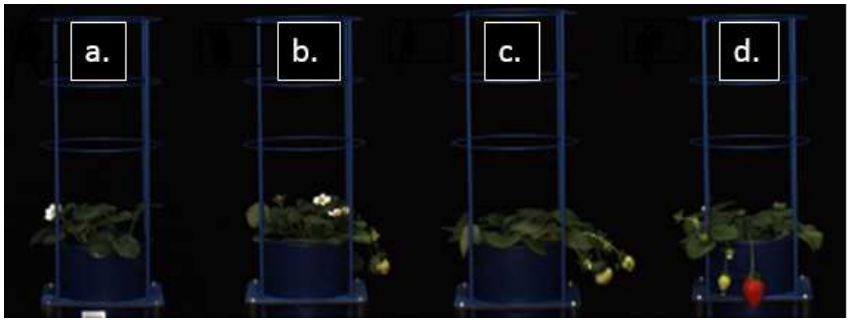


Fig. 6.1 Example pictures from the dataset: a. Flowering stage, b. Turning flowers into fruits, c. Unripe fruit stage, d. Ripe fruit stage

Annotation was performed using the CVAT tool, defining four bounding box categories: flowers, young fruits, medium-sized fruits, and ripe fruits. The dataset was randomly divided into training and validation sets with a ratio of 8:2. Table 6.1 presents the number of samples per class for training and validation subsets and class imbalance in the training set through the label distribution. This disparity has the potential to lead to the model learning overfitting the flower class, due to the more prominent representation. To address this problem, these classes were first merged into two broader categories: flowers and fruits, thus partially unifying the data. It is necessary to recognize the inherent variation in the size and color of fruits and flowers. Because the dimensions of the fruits fluctuate from image to image and in the same image, and the color palette spans the gamut from yellow, to green and red. This variability can be attributed to the inclusion of strawberries at various stages of growth, which in turn contributes to the distinctive visual characteristics exhibited by both fruits and flowers. This study implemented data augmentation methods to enhance the model capacity for generalization and mitigate the risk of overfitting during training (flip, scale, and translate).

Table 6.1 Number of instances per class in train and validation.

Categories	#	Train	Val
Four classes	Total	4303	1138
	Young fruit	1457	382
	Medium fruit	698	182
	Mature fruit	279	54
	Flower	1869	520
Two classes	Total	4303	1138
	Fruit	2434	618
	Flower	1869	520

6.2.2 YOLOv8 Architecture

YOLOv8, developed by Ultralytics (Ultralytics, Maryland, USA), represents an advancement, building upon the achievements of its precursor versions. It achieves state-of-the-art performance by employing a combination of techniques, including structural optimization, anchor box refinement, anchor-free strategies, and a diverse array of data augmentation methods. Introducing a versatile selection of model sizes, ranging from nano and small to middle, large, and extra-large, YOLOv8 offers a comprehensive toolkit for researchers. In the deep learning architecture, the decision was made to predominantly base it on the YOLOv8l (large-size variant). As demonstrated in our previous research [24], larger models consistently exhibit superior performance compared to their smaller counterparts. However, their challenge is in the increased time and memory requirements for training these models, which can be a costly endeavor. YOLOv8 comprises three integral components: the backbone, neck, and head, each serving a distinct purpose in the process of feature extraction, multi-feature fusion, and prediction output. The backbone network utilizes convolutional operations to capture multi-scale features from images. The network goes through five downscaling stages, resulting in the creation of five distinct layers of feature representations denoted as P1, P2, P3, P4, and P5. Each Pi represents a resolution that is a fraction of 1/2i compared to the original image. The input image undergoes downscaling with 8, 16, and 32 from layers p3, p4, and p5, respectively,

and creates multiple layers of feature maps. Simultaneously, the neck network plays an essential role in amalgamating the features extracted by the backbone network. It is essential to note that the process of multi-scale feature fusion within the neck network does not change the scale of the feature maps. These feature maps are subsequently fed to the detection head, which meticulously processes them at three specific scales: 20×20 (small), 40×40 (medium), and 80×80 (large), all while preserving the input image size at 640×640. The head layer takes charge of predicting the target categories, and the detection task is accomplished through the utilization of these three sets of detectors with varying sizes to identify and categorize the contents within the images. Based on our data, the objects show a range of sizes and shapes, which can impact the models recognition capabilities. For example, young fruits are small in size and closely resemble the background (leaves) in terms of color potentially causing disruptions in model recognition. Conversely, medium and ripe fruits can divert the models attention away from smaller objects. Therefore, to obtain a more comprehensive level of detail, we decided to adjust the image sizes to 960×960. This modification was deemed essential due to the constrained perspective of the objects in focus, specifically young fruits and small flowers. Since the main responsibility of processing the feature maps from P3, P4, and P5 layers with the head of the detection, The focus of this paper was centred on optimizing the model head to achieve more precise object detection, particularly for smaller objects. The basic structure of the YOLOv8 model is shown in 6.2.

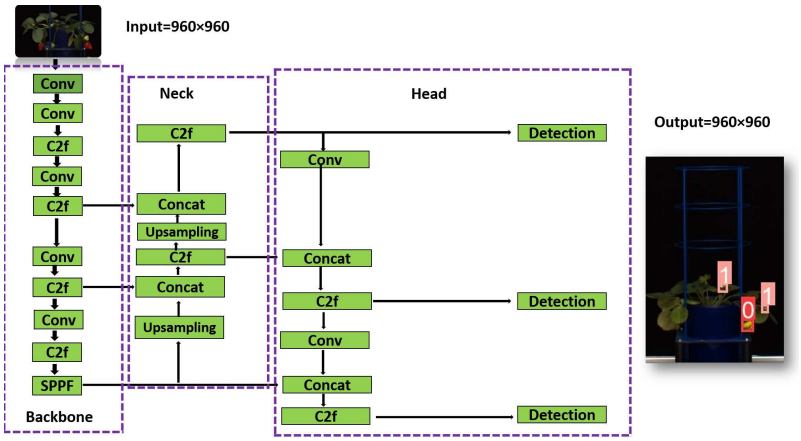


Fig. 6.2 The architecture of the YOLOv8 model

6.2.3 Optimizing YOLOv8 Model Head

To optimize the model head, two techniques were employed:

- Adding a small object detection head from the P2 layer on the YOLOv8 model enhances the ability of the model to detect smaller objects, then deleting the P5 layer as a large object detection.
- Implement the same approach by adding a SE-Block attention module.

Identifying small and young fruits and flowers in the images, typically sized at less than 10×10 pixels is an important challenge. After multiple rounds of down-sampling, these marks lose their distinctive features, posing a persistent challenge for precise detection at high resolutions using the P3 layers detection head. Due to the limited feature representation of small targets, intricate details, and precise location information are gradually reduced as the downsampling process progresses within the backbone network. This excessive downsampling can ultimately lead to the loss of small targets. On the other hand, shallower layers uphold a greater spatial resolution, safeguarding essential positions and intricate details, which makes them especially adept at detecting smaller targets. Faced with this challenge, the decision was made to bolster the capabilities of the YOLOv8 lightweight network architecture. This method involves adding the P2 layer to precisely locate small flowers and fruits while retaining their essential texture features. This addition has been meticulously engineered to safeguard the complex texture characteristics associated with these small-scale objects. The updated framework of the model with four detection heads is shown in Figure 6.3.

In this new head, the P2 layer detection head functions with a resolution of 240×240 pixels, equivalent to only two down-sampling stages in the backbone network. This resolution retains more extensive information regarding the underlying target features. The two P2 layer features, originating from both top-down and bottom-up processes within the neck network, are seamlessly integrated with the corresponding scale features from the backbone network through concat. Therefore, the P2 layer detection head operates swiftly and effectively when tasked with detecting small targets, as the output features are the result of fusing all three input features. This head is meticulously designed to function with underlying features and is composed of high-resolution, low-level feature maps that show exceptional sensitivity when

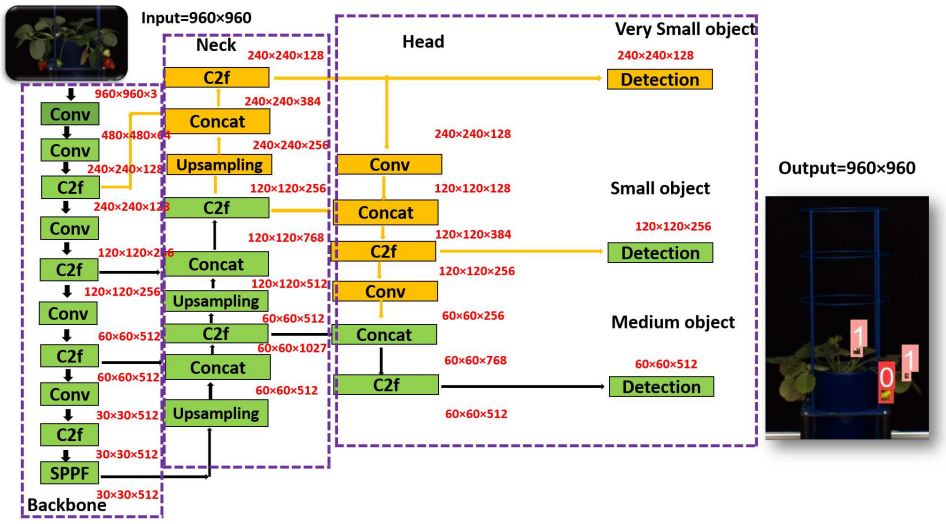


Fig. 6.3 Structure of YOLOv8 model with adding a detection head (p2 layer) (orange color)

detecting smaller objects. Every layer within the model is tasked with object detection at specific scales. Therefore, since our data covers mostly small objects, this idea can have a positive effect. However, the process began by removing the P5 layer and then training the data with this method.

On the other hand, an attempt was made to enable the network to effectively retune its features by adding the SE-Block attention module to the P2 layer (as it was useful in previous work). It makes this selectively recognize valuable features using enhanced global information while reducing the importance of less relevant items. The objective of this module is to promote the significance of essential information about the classes (fruits and flowers) and amplify their weight, therefore enhancing their distinctiveness and precision. Figure 6.4 shows the structure of the P2-YOLOv8 model with the SE-Block attention module.

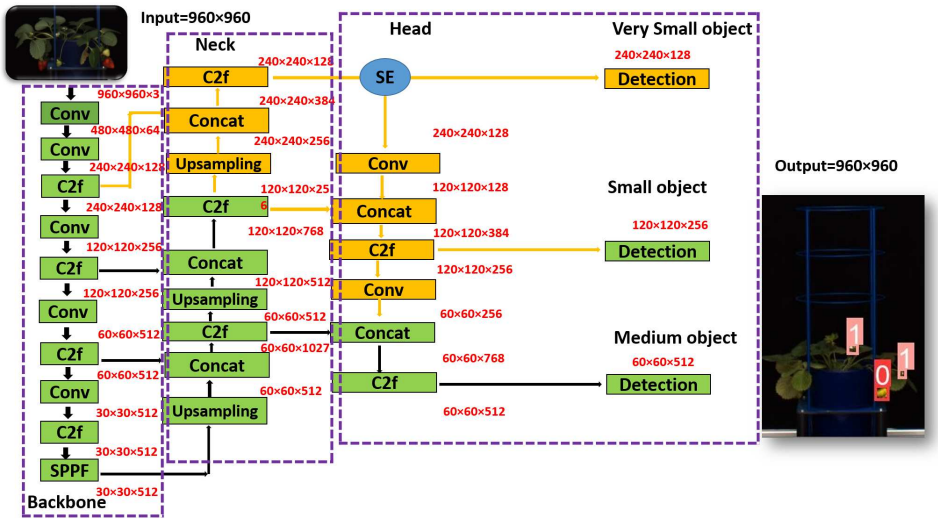


Fig. 6.4 Structure of P2-YOLOv8 model with SE-Block attention module (Blue color)

The SE block, introduced by [83], stands out as an exceptionally effective attention mechanism designed to explore channel relationships. This mechanism empowers the network to efficiently recalibrate its features, allowing for the selective amplification of significant features through the integration of global information, while simultaneously reducing the prominence of less pertinent ones [122].

6.2.4 SO-YOLOv5 with Replacing the YOLOv8 Head

In pursuit of enhancing the precision of strawberry flower and small fruit detection, the SO-YOLOv5 model [219] was used, which is derived from the YOLOv5s model (this model was not open source, so code was written independently). However, it is imperative to note that in the context of this research endeavor, the established YOLOv5 framework was deviated from, and the YOLOv8l model was employed. This decision was made because YOLOv8 utilizes an advanced detection head. Therefore, the YOLOv8 model head was exchanged with the SO-YOLOv5 model head due to its improved detection capabilities. In YOLOv8, it shifts from the connected structure of YOLOv5 to a separate configuration. YOLOv8 is characterized as an anchor-free model, while YOLOv5 adheres to an anchor-based approach. YOLOv5 relies on predicting bounding box locations and sizes relative to anchor boxes, with subsequent adjustments based on the offset between the anchor boxes and the ground truth boxes.

This methodology facilitates the models capability to accurately identify objects of varying sizes and proportions. Conversely, anchor-less models like YOLOv8 dispense with anchor boxes altogether. Instead, these models directly predict the center points and sizes of bounding boxes, reducing model complexity and obviating the manual specification of anchor boxes. In Figures 6.5 and 6.6, the detection heads of the YOLOv5 and YOLOv8 models can be observed, respectively.

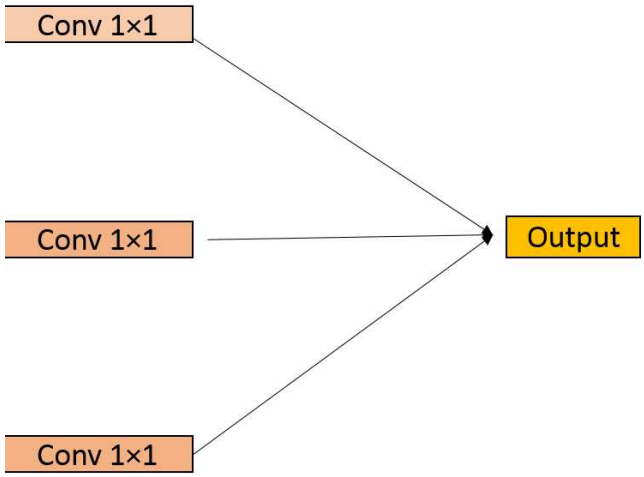


Fig. 6.5 detection head of YOLOv5

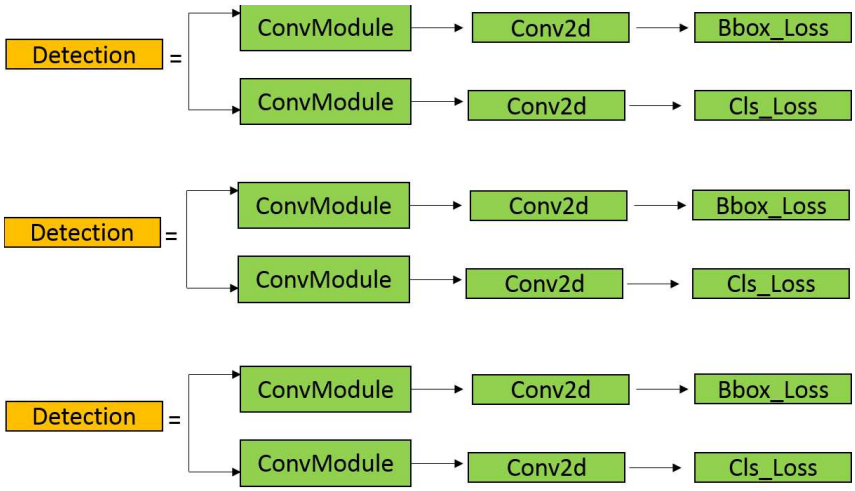


Fig. 6.6 detection head of YOLOv8

SO-YOLOv5 model employs three distinct operations to improve small object recognition. It employs a feature pyramid network called BiFPN in the neck section to enhance the visualization of features extracted from various backbone layers,

ensuring that essential strawberry characteristics are not lost. In the subsequent step, it was designed a small object detection head at the p2 layer for recognizing small objects with a detection size of 240x240. This resolution meticulously preserves a comprehensive set of details related to small fruit features. Lastly, it was integrated the CA (Coordinate Attention) mechanism within the feature fusion layer, effectively emphasising the feature information essential for accurate strawberry recognition. The structure of the SO-YOLOv5 model and The structure of CBS, CSP1, and csp2 modules can be seen in Figures 6.7 and 6.8, respectively.

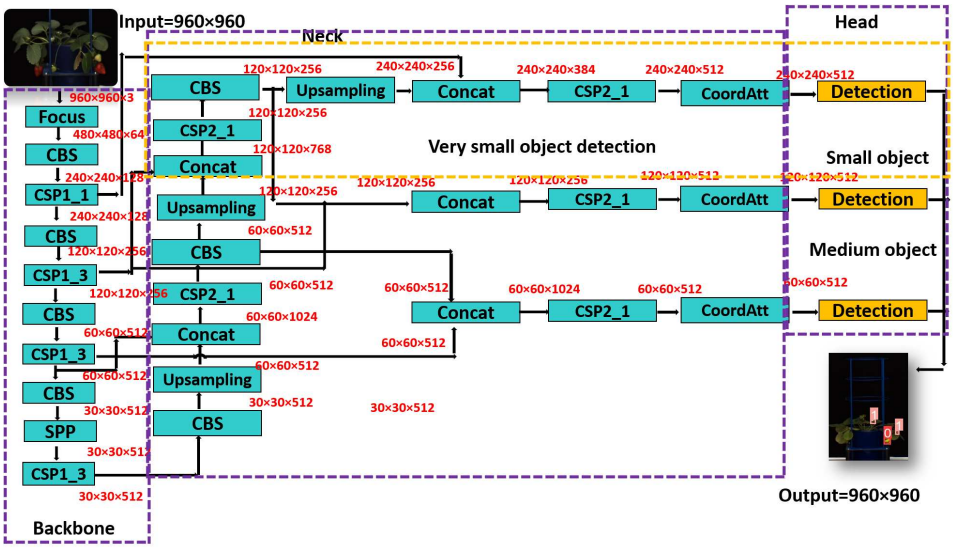


Fig. 6.7 The architecture of the SO-YOLOv5 model

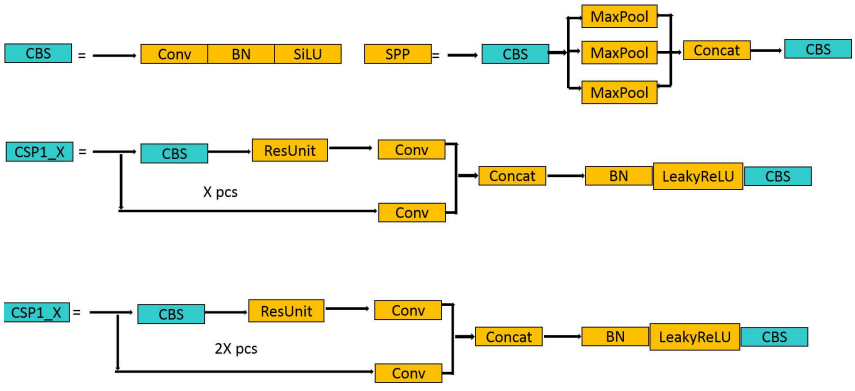


Fig. 6.8 The structures of CBS, CSP1, and csp2 modules

Following feature extraction from the backbone network, the incorporation of both high-level and low-level features is crucial for enhancing the algorithms effectiveness in detecting strawberries. YOLOv5 employs PANETs bidirectional feature fusion technique to merge these features, thereby promoting feature integration and utilization. Nevertheless, it falls short of accurately identifying crucial feature contributions and imposes overhead in terms of parameters and computational resources [105], [219]. BiFPN, an acronym for Bidirectional Feature Pyramid Network, stands as a remarkably efficient architectural innovation. It simultaneously establishes both top-down and bottom-up channels, enabling seamless feature integration across diverse scales. Furthermore, it incorporates horizontal connections between features within the same scale, effectively addressing the potential loss of valuable feature information inherent in deep network layers [191]. A standout feature of BiFPN is its capacity to efficiently recycle its feature network layer multiple times.

Going beyond conventional structures, this design introduces learnable weights to evaluate the significance of various input features. Through a judicious combination of top-down and bottom-up fusion processes, it merges multi-scale features, resulting in an enhanced equilibrium of feature information across varying scales. The mathematical formulation that characterizes the fusion process in BiFPN is delineated through the equations:

$$P_{i^{td}} = \frac{\omega_1 \cdot P_{i^{in}} + \omega_2 \cdot \gamma(P_{i+1^{in}})}{\omega_1 + \omega_2 \cdot \varepsilon} \tag{6.1}$$

$$P_{i^{\mathrm{Out}}} = \partial\left(\frac{\omega_1' \cdot P_{i^{\mathrm{in}}} + \omega_2' \cdot P_{i^{\mathrm{td}}} + \omega_3' \cdot \gamma(P_{i-1^{\mathrm{Out}}})}{\omega_1' + \omega_2' + \omega_3' + \varepsilon}\right) \tag{6.2}$$

Detecting small and immature fruits and flowers within images poses a significant challenge. To address this issue, SO-YOLOv5 introduces additional up-sampling and down-sampling procedures by integrating the concat layer, convolution layer, and CSP module [219]. In this approach, the concat layer is utilized to amalgamate feature layers of the same scale, encompassing those with smaller object feature information sourced from the backbone network [111]. The feature map scales generated by the ultimate recognition layer are 60×60, 120×120, and 240×240, respectively.

To discern the pivotal feature information within input image data for a specific task, attention mechanisms play a vital role. This model introduces the Channel Attention (CA) mechanism to address the challenge of low detection accuracy, particularly with smaller objects (Small flowers and small fruits). It is purpose is to enhance the precision of object-related feature extraction while mitigating the influence of extraneous data from various channels. Figure 6.9 shows a coordinate module mechanism.

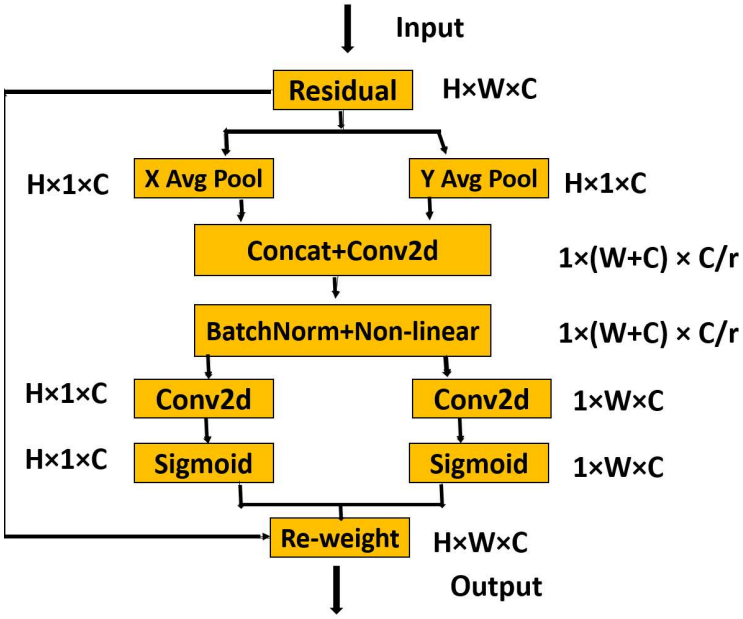


Fig. 6.9 Coordinate attention mechanism

6.3 Experimental Results and Discussion

This study extends the lightweight network architecture YOLOv8 by incorporating a P2 layer and removing the P5 layer to enhance the detection of small objects in strawberry plants. Subsequently, a similar strategy was followed by integrating a Squeeze-and-Excitation (SE) block attention module into the main part of the model (on the P2 level) to emphasize crucial information about classes such as fruits and flowers. The SO-YOLOv5 model was implemented by replacing its head with the YOLOv8 model head to reduce model complexity and eliminate manual specification of anchor boxes. Finally, the performance of the basic YOLOv8 model was evaluated against each approach (P2-YOLOv8 and SO-YOLOv5).

This section is organized into five parts:

- The first part explains zero-shot evaluation and transfer learning.
- The second part examines the standard YOLOv8 architecture.
- The third part describes the YOLOv8 architecture with the P2 layer.
- The fourth part discusses the P2-YOLOv8 architecture with the SE-block module.
- The fifth part explores the impact of SO-YOLOv5 with the YOLOv8 head architecture.

6.3.1 Evaluation from Scratch and Transfer Learning

The YOLOv8l model underwent training with a comprehensive dataset comprising fundamental and augmented data in this study. This training was executed using both from scratch and transfer learning approaches, elucidating a multifaceted methodological framework. It is necessary to mention that, consistent with the previous study [187], the YOLOv8 model was trained with the SE-Block attention module on the tomato dataset to identify flowers, fruits, and nodes. The objective is to use the transfer learning method to transfer the best weight of the tomato dataset trained on the strawberry dataset. Therefore, in this study, the research was conducted through the exploration of three distinct approaches, namely, from scratch, pre-training YOLOv8l, and pre-training tomato.

6.3.2 YOLOv8 Standard Architecture

The results in Table 6.2 show a discernible trend as the transition is made from scratch training to transfer learning. Specifically, the precision metric shows improvement of approximately 1.86%, advancing from 86.76% during the scratch stage to 88.37% with pre-trained YOLOv8l weights. This emphasizes the significance of knowledge transfer through pre-trained YOLOv8l weights, resulting in more accurate predictions. Applying the pre-trained tomato weights to the strawberry dataset yields higher precision than scratch by 1.73%. Nevertheless, there is a slight decreasing trend compared to the pre-trained YOLOv8l weight of about 0.12%. Examining the

F1 score values in Table 1, it is evident that the highest F1 score is associated with the pre-training tomato weight with augmentation at 87.96%. With the integration of data augmentation, a consistent and incremental improvement was observed in the scratch training procedure. However, during the transfer learning process, applying the same data augmentation techniques led to a decrease in precision for both pre-training YOLOv8l and pre-training tomato. During transfer learning, the model has already acquired specific patterns and features from the source dataset (In this case, pre-trained weights). When data augmentation techniques are applied at this stage, they may introduce noise or conflicting information. The model may have already developed a certain understanding of the data, and abrupt changes due to augmentation might impede it is the ability to adapt effectively to the new dataset. This can result in a decline in precision, indicating that the model is less accurate in making positive predictions, such as fruit and flower detection. However, it is worth noting that data augmentation was effective in increasing the ability of the model to identify all instances of objects in images (Recall).

Table 6.2 The evaluation of the YOLOv8l standard from scratch and transfer-learning on the strawberry dataset with and without augmentation

Training approaches	P%	R%	F1	mAP50%
Scratch	86.76%	88.21%	87.47%	92.31%
Scratch(Augment)	**88.38%**	86.74%	87.55%	92.49%
Pre-training(l)	88.37%	86.44%	87.39%	92.01%
Pre-training(l)(Augment)	86.30%	86.70%	86.49%	92.19%
Pre-training tomato	88.26%	86.95%	87.60%	92.23%
Pre-training tomato(Augment)	87.30%	**88.65%**	**87.96%**	**92.52%**

6.3.3 YOLOv8 Architecture with P2 Layer

let us advance the discussion by amplifying the YOLOv8 architecture by incorporating a small object detection layer denominated as the P2 layer. As described in section C, our primary objective is to improve the accuracy of detecting small objects

(Especially flowers and small fruits) and increase the performance of the model by adding the P2 layer. Table 6.3 reveals that the pre-training tomato approach yields the highest precision with 90.20%. However, when applying augmentation to this approach, there is a decrease in model precision of about 0.99%. The lowest recorded precision of 85.69% pertains to the scratch approach with utilizing augmentation. Therefore, including the P2 layer during the training process with the pre-training tomato approach and data augmentation positively impacts the model performance. This is evidenced by an increase in both F1 and mAP50% scores, registering at 88.26% and 93.24%, respectively.

Table 6.3 The evaluation of the P2-YOLOv8l without P5 from scratch and transfer-learning on the strawberry dataset with and without augmentation

Training approaches	P%	R%	F1	mAP50%
P2-Scratch	87.64%	86.99%	87.31%	92.14%
P2-Scratch(Augment)	85.69%	**88.28%**	86.96%	92.96%
P2-Pre-training(l)	88.30%	88.10%	88.19%	92.30%
P2-Pre-training(l)(Augment)	88.38%	87.89%	88.13%	93.22%
P2-Pre-training tomato	**90.20%**	85.25%	87.65%	92.03%
P2-Pre-training tomato(Augment)	89.30%	87.26%	**88.26%**	**93.24%**

However, it is essential to note that this approach had contrasting outcomes without data augmentation which resulted in a reduction of F1 and mAP50% scores by about 87.65% and 92.03%, respectively.

6.3.4 P2-YOLOv8 Architecture with SE-block Module

An attempt was made to enable the network to effectively retune its features by adding the SE-Block attention module to the P2 layer and after the C2f module (as it was useful in previous work on tomatoes)[187]. After applying the SE-Block module, the results depicted in Table 6.4 show a decrease in the performance of the pre-training tomato approach. This decrease is obvious when compared to pre-

training YOLOv8l, which shows the highest precision with 90.45%. It is possible that training the data with pre-trained tomato weights, particularly when overlaid with SE-Block modules, leads to a disruption in the training process. The likely reason for this disruption could be the conflicting or redundant information present in both the pre-trained tomato weights and the new SE-Block module. When the model is being trained, these overlapping components may introduce inconsistencies or confusion, hindering the learning process. The clash between the information encoded in the pre-trained tomato weights and the additional attention mechanism introduced by the SE-Block module might cause the model to struggle to adapt effectively to the specific features of the new dataset. This interference could lead to suboptimal performance or slower convergence during the training phase. By applying augmentation on the pre-training YOLOv8l, the highest performance was achieved with mAP50% of 93.50% and an F1 score of 88.26%.

Table 6.4 The evaluation of the P2-YOLOv8l with SE-Block from scratch and transfer-learning on the strawberry dataset with and without augmentation

Training approaches	P%	R%	F1	mAP50%
SE-Scratch	87.59%	86.99%	87.28%	92.46%
SE-Scratch(Augment)	85.80%	88.41%	87.08%	93.33%
SE-Pre-training(l)	**90.45%**	86.15%	88.24%	92.29%
SE-Pre-training(l)(Augment)	86.74%	**89.84%**	**88.26%**	**93.50%**
SE-Pre-training tomato	90.29%	85.39%	87.77%	92.47%
SE-Pre-training tomato(Augment)	88.29%	87.83%	87.83%	92.88%

6.3.5 SO-YOLO5 with YOLOv8 Head Architecture

To enhance the precision of identifying small flowers and strawberries, as outlined in section D, this study introduced the SO-YOLOv5 model. As depicted in Table 6.5, the pre-training tomato approach achieved the highest recognition precision with 90.38%. However, the pre-training YOLOv8l approach obtained the highest F1 score at 89.07% by increasing recall.

Table 6.5 The evaluation of the SO-YOLOv5 from scratch and transfer-learning on the strawberry dataset with and without augmentation

Training approaches	P%	R%	F1	mAP50%
SO-Scratch	87.77%	87.18%	87.47%	**93.42%**
SO-Scratch(Augment)	87.64%	89.40%	88.51%	92.97%
SO-Pre-training(l)	87.68%	86.86%	87.26%	93.16%
SO-Pre-training(l)(Augment)	88.15%	**90.01%**	**89.07%**	93.29%
SO-Pre-training tomato	**90.38%**	85.76%	88.00%	92.65%
SO-Pre-training tomato(Augment)	85.91%	88.49%	87.18%	93.31%

6.4 Summary

Strawberry fruits, despite their small size, possess a remarkable ability to seamlessly blend into their foliage surroundings. This size differential presents significant challenges even for advanced deep neural network (DNN) models in accurately identifying them.

In light of this, the current chapter delves into the exploration of YOLOv8, aiming to tackle the complexities of a dataset gathered through a high-throughput phenotyping platform. The study proposes a thorough investigation into optimized adaptations of the model head to improve the detection of small objects, starting with the foundational YOLOv8 architecture. This optimization is pursued while ensuring the model efficacy in detecting larger fruits and flowers, poised to yield enhanced detection accuracy and robust results on the specified dataset.

Achieving the optimal balance between model complexity and performance remains a daunting task. Hence, the primary aim of these experiments is to enhance the adaptability of the YOLOv8 model head specifically for the detection and performance improvement of smaller strawberry fruits and flowers, minimizing additional layers and parameters.

Results indicate that in the standard YOLOv8l model, leveraging pre-trained tomato weights and applying data augmentation yields the best performance. Notably, integrating the shallower layer P2 while excluding the deeper layer P5 enhances the preservation of intricate details in small strawberry flowers and fruits. Although this study primarily focuses on modifying the YOLOv8 model head, exploring the SO-YOLOv5 model with a similar head structure to YOLOv8 demonstrates superior results by adapting both the backbone and neck of the model. Ultimately, these adaptations highlight the considerable potential to enhance YOLOv8 detection capabilities, especially for diminutive botanical specimens such as strawberries, by carefully refining model architecture and training approaches.

Chapter 7

Identification of Plant Roots Using Convolutional Neural Networks

7.1 Overview

In the current era, marked by rapid advancements in science and technology, the study of plant roots has become essential for improving crop shapes, breeding new varieties, predicting climate change, and optimizing cultivation techniques. Our approaches to researching root systems have evolved significantly, transitioning from destructive to non-destructive methods, and from traditional manual in situ root segmentation to automated techniques. These more efficient technologies offer a faster route to analyzing root phenotypic characteristics, enhancing the overall understanding and management of plant growth.

Plant phenotyping aims to conduct non-destructive analysis of complex plant traits related to growth, yield, and adaptation to stress with high accuracy and precision. Traditionally, these tasks are carried out by human operators whose performance may be limited by their experience and skills. Moreover, the advent of several High-Throughput Platforms (HTPs) in recent years has generated a substantial amount of data on plant phenotypical traits, leading to increased workloads for human operators. This has highlighted the need for automatic and unbiased protocols to manage and analyze the data effectively.

RSAs are difficult to observe directly, mainly due to the soil which naturally covers them [141]. As a consequence, specific non-destructive phenotyping methods have been developed, such as the use of transparent agar or germination papers, which have proven to be efficient, especially at early growth stages, with the only disadvantage of requiring the root system to grow in artificial soil [159]. Another viable approach is the use of X-ray computed tomography [155], which allows the visualization of the root system in natural soil; however, this type of system is expensive and difficult to deploy directly on the field.

Once images of the RSA have been gathered, they should be segmented to detect the roots. However, the segmentation step is usually challenging due to the complex nature of the RSA and the low contrast between soil particles and roots. To cope with these issues, several tools have been proposed. As an example, the authors in [166] proposed a framework named *GLO-Roots*, which exploits different types of feature-based image analysis techniques, such as local pattern recognition, global, shape, and directionality analysis, to identify and extract the characteristics of the root system, also considering gene reporters and soil moisture. Another semi-automated tool, called *GT-Roots*, is proposed in [17]; GT-Roots also applies a processing pipeline to each image that starts by extracting a Region of Interest (RoI), then converts the original image into grayscale, performs adaptive thresholding, and finally applies a morphological operator to enhance the results. GT-Roots also allows for a semi or fully-automated pipeline, where the operator can manually intervene in each intermediate processing step. Authors in [62] propose *GIA-ROOTS*, whose pipeline first performs image pre-processing via rotation, crop, and scaling. Afterward, the user is asked to select a series of relevant root system traits from a set of 19 possible choices, which are then used on the segmented image to extract the root system. Another tool is *saRIA* [148], which provides a semi-automated environment for RSA segmentation and calculating phenotypic features of the RSA. The analysis pipeline includes several pre-processing steps, such as cropping, despeckling, smoothing, and inversion of image intensity. Then, adaptive image thresholding is used to segment the image in the foreground (roots) and background (soil) using Gaussian weighted mean. Morphological filters are then used to remove noise and improve the quality of the found roots, root skeletons are computed, and RSA features are extracted from a list of 44 root traits using a pixel-wise computation. While these tools can perform extremely well when only a few images are available, they may be inadequate when a high amount of data must be processed due to the required human

intervention. Furthermore, fixed processing pipelines usually lack generalization capabilities. Tools based on deep learning have been proposed to deal with these issues. One, and probably the most well-known, of such tools is SegRoot [207], that provides a binary mask of root (white pixels) and no-root (black pixels) starting from an RSA image. SegRoot is based on a modification of SegNet [10] and uses a series of standard CNN blocks (that is, 3×3 convolutional filters followed by batch normalization and ReLU activations) in the encoder. The decoder is composed of a series of unpooling layers that perform non-linear upsampling to make the output feature maps identical to the input feature maps of the corresponding encoding layer. The main difference between SegRoot and SegNet lies in the loss function, which is a modification of the Dice coefficient [134]. Another tool based on a U-architecture is DeepLabv3+ [181], which uses an U-shaped encoder based on Xception as its backbone. The approach proposed in [193] predicts two parameters representing the vertical and horizontal centroid of root distribution to reveal the phenotypic diversity of root distribution.

This chapter introduces a novel approach for identifying plant roots from images of root system architectures using a convolutional neural network (CNN). The CNN processes small image patches to calculate the probability that the center point of each patch is a root pixel. The core idea is to ensure that the CNN model captures as much information as possible about the variability within patches, which often feature chaotic and heterogeneous backgrounds. Results on a real dataset demonstrate the feasibility of this approach, showing that it surpasses the current state of the art in RSA segmentation.

7.2 Methodology of identification of plant roots

Although effective, U-shaped models can be overly complex for classification tasks and typically yield only binary outputs, failing to evaluate the probability of observing root versus no-root regions in an image. To address this limitation, we introduce a comprehensive processing pipeline for the end-to-end analysis of Root System Architectures (RSAs) in plants. This pipeline begins with the automatic extraction of image patches from labeled images. The pixel-based extraction criterion produces a large number of patches from a relatively small set of RSA images, thereby reducing the data collection and labeling workload for human operators.

The end-to-end analysis of RSAs presented in this chapter includes several key components:

1. Image Data Acquisition and Annotation: A review of the methods used to acquire and annotate the image data.

2. Data Preprocessing: An outline of the preprocessing steps applied to the data.

3. Performance Analysis of the RootNet Architecture: A discussion on the performance of the RootNet architecture.

7.2.1 Dataset gathering and annotation

Data collection spanned several days using cylindrical rhizotron tubes that housed plants of a specified genotype. This process yielded numerous snapshots, each comprising multiple photos of a rhizotron tube, resulting in approximately [specific number] raw RGB images. These images were captured using a Basler AG Scout sca1600-14gc camera, with a spatial resolution of 1234 × 1624 pixels (width × height). To minimize environmental variables that might impact image quality in uncontrolled settings—such as outdoor environments with varying lighting conditions or using different sensors to capture similar data—data collection was conducted using the High Throughput Plant Phenomics Platform (HTP), based on the LemnaTec Scanalyzer3D system at the ALSIA Metapontum Agrobios Research Centre.

The raw images were subsequently processed, beginning with the identification of the rhizotron border. This border was then rotated to determine the cylinder radius using geometrical formulas. Following this, the images were stitched together to complete the panorama extraction. Finally, noise resulting from light reflection on the cylindrical rhizotrons was removed using Singular Value Decomposition (SVD).

Images from the original dataset were selected for manual data annotation. To ensure the accuracy and consistency of the annotated ground truth data, three independent domain experts labeled the ground truth masks. A reliability check was then conducted using a majority voting procedure. Specifically, a pixel in the ground truth mask was labeled as root (or non-root) if at least two of the three annotators

agreed on the classification. This approach aimed to minimize subjective bias that could arise from a single annotator, even an expert. An example of a ground truth image and its corresponding annotation is shown in Figure 7.1.

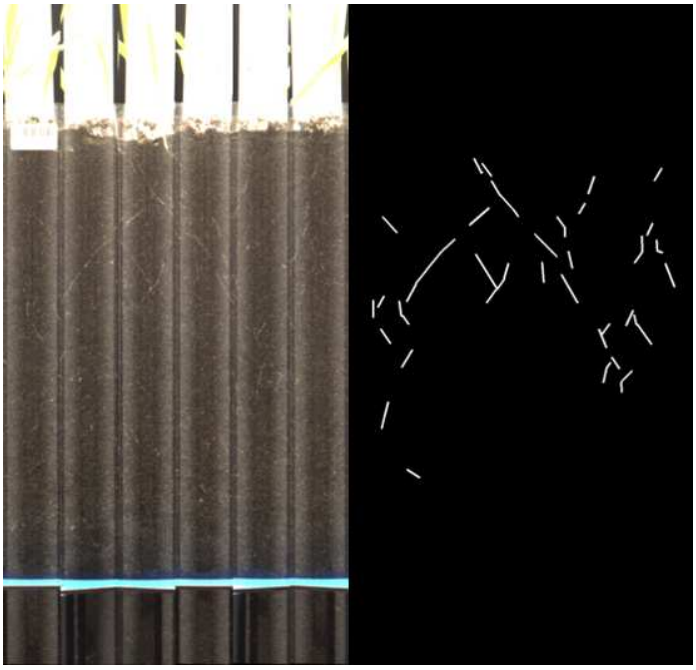


Fig. 7.1 On the left, a root composite image with its corresponding ground truth on the right.

7.2.2 Data preprocessing

Let us note that the raw annotated images are not directly utilized in the experiments.

Instead, image patches have been automatically created using an *extraction filter* of size $F_w \times F_h$. Specifically, the filter "flows" across each pixel in the image, starting from the top left towards the bottom right, provided that:

$$x_P \in \left[I_w + \frac{F_w}{2}, I_w - \frac{F_w}{2} \right] \wedge y_P \in \left[I_h + \frac{F_h}{2}, I_h - \frac{F_h}{2} \right] \tag{7.1}$$

In equation 7.1, (x_P, y_P) represents the coordinates of the pixel P, while I_w and I_h represent the width and the height of the original image, respectively. In other words, the filter extracts a series of patches provided that its borders completely fit within the original image, hence not applying any padding operation.

After extracting each patch, its corresponding ground truth was used to automatically classify it as either a positive sample (where its center represents a root pixel) or a negative sample (where its center does not represent a root pixel).

It's important to acknowledge that datasets generated this way can be highly imbalanced, often containing a larger number of negative samples. To address this imbalance during the automatic extraction of image patches, a specific number of root patches were first extracted from each image. Subsequently, an equal number of no-root patches were randomly selected by subsampling the background of the image. This approach ensured that approximately 250,000 patches were collected for each class, as illustrated in Figure 7.2.

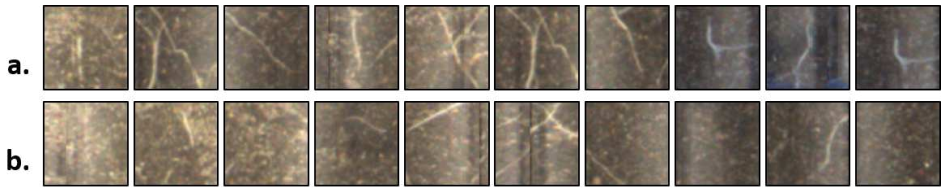


Fig. 7.2 RootNet dataset patches arranged in two rows: row a. that shows examples from the root class and row b. that shows examples from the non-root class. All the patches must span the highest number of background configurations possible to consider root image complexity. A non-root patch can have roots in the surroundings, but the center point must be a non-root.

In preparation for training RootNet, a data augmentation step was implemented. This involved randomly rotating images around their center points, as well as flipping them horizontally and vertically. Additionally, adjustments were made to enhance sharpness, brightness, contrast, or saturation. Importantly, these augmentation operations preserved the original class of each patch, as they did not affect the center point. No color distortions were introduced to prevent the inclusion of improbable data in the dataset.

7.2.3 RootNet architecture

In the conducted experiments, an efficient CNN-based architecture named RootNet was devised. RootNet consists of three consecutive CNN layers with max pooling and ReLU activation. The design principles outlined in [77] were adhered to, which involve doubling the number of convolution filters when reducing the feature map size. Following the convolution layers, a fully connected layer was employed,

followed by a sigmoid activation function. The choice of sigmoid over softmax was motivated by framing the problem as a binary classification task, where RootNet determines whether each patch represents a positive or negative sample. For training, binary cross-entropy was utilized as the loss function, optimized using SGD. A schematic overview of RootNet architecture is presented in Figure 7.3.

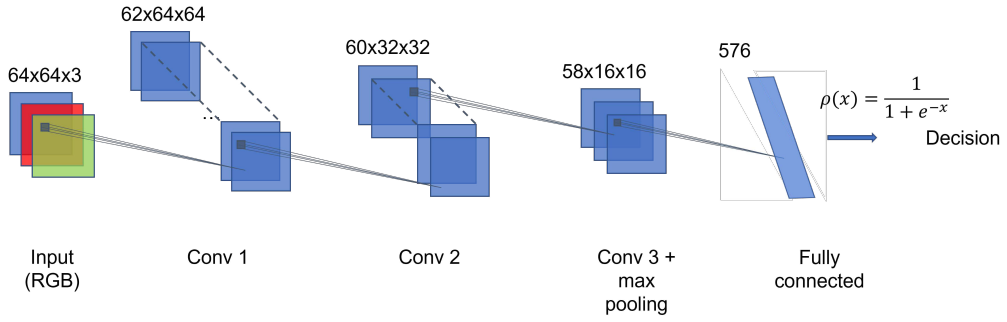


Fig. 7.3 RootNet architecture. The proposed architecture sends the RGB image through three different convolutional layers, with a decreasing density of the applied kernels. After the third convolution, a max pooling layer is applied to retain relevant features, which are then fed to a fully connected layer and, finally, to the decision layer.

7.3 Experimental Results and Discussion

In this section, the performance of the RootNet architecture was first evaluated by varying the input image size and measuring precision, recall, accuracy, and F1 score. Next, the results of RootNet were compared with those of SegRoot, highlighting the fundamental differences in their architectures: SegRoot utilizes a U-shaped encoder-decoder network, while RootNet employs a stacked set of convolutional, ReLU, and max-pooling layers, topped with a binary classifier. Finally, a pixel-based quantitative comparison was proposed to further assess the performance of RootNet against SegRoot.

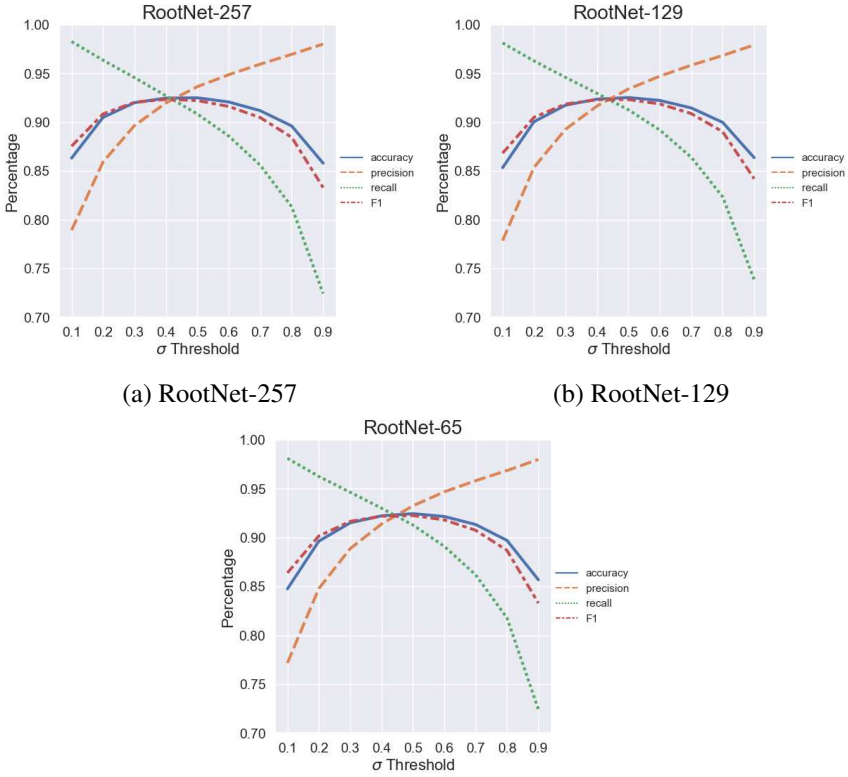
7.3.1 RootNet performance

First, the results of the RootNet architecture have been evaluated by varying the input image size in terms of precision, recall, accuracy, and F1 score. Specifically, three models with three different image input sizes have been used, that is, 257×257 (i.e.,

Table 7.1 Metrics achieved by RootNet at a fixed value of $\sigma = 0.45$ after data augmentation.

Model	**A** (%)	**P** (%)	**R** (%)	**F1** (%)
RootNet-257	**92.47%**	**91.97%**	92.71%	**92.34%**
RootNet-129	92.36%	91.65%	92.96%	92.30%
RootNet-65	92.22%	91.38%	**93.00%**	92.18%

RootNet-257), 129×129 (i.e., *RootNet-129*), and 65×65 (i.e., *RootNet-65*). From these networks, the raw value predicted by the binary classifier has been extracted, that is, the raw value extracted by the sigmoid activation function. Several fixed values for the σ threshold are used to compute evaluation metrics. The results are reported in Figure 7.4a, 7.4b and 7.4c. Furthermore, the numerical values achieved by the metrics at the threshold of $\sigma \sim 0.45$ are shown in Table 7.1.



(a) RootNet-257

(b) RootNet-129

(c) RootNet-65

Fig. 7.4 From left to right, evaluation of Accuracy, Precision, Recall, and F1-score for RootNet-257, RootNet-129, and RootNet-65 at σ threshold levels from 0.1 to 0.9, sampled with a step of 0.1.

Table 7.2 shows the classification results achieved by the available configurations of RootNet without data augmentation. The results clearly show how data augmentation improves the overall performance of the networks. Interestingly, this is mainly related to a lower recall achieved by the network when trained on non-augmented data, resulting in a decrement in the performance of the network in the correct identification of root patches. This could be ascribed to the augmentation steps that keep the central point belonging to a root (or no-root), that increase the variability of the observed scene, resulting in better performance.

Table 7.2 Metrics achieved by RootNet at a fixed value of $\sigma = 0.45$ without augmentation.

Model	**A** (%)	**P** (%)	**R** (%)	**F1** (%)
RootNet-257	86.40%	**98.10%**	73.62%	84.12%
RootNet-129	86.18%	97.98%	73.44%	83.96%
RootNet-65	**87.81%**	97.95%	**76.89%**	**86.15%**

The first thing to notice is that the precision value shows direct proportionality with the σ threshold. In contrast, the recall shows inverse proportionality to the same threshold. As a consequence of this behavior, the accuracy and F1 curves show an inverted U-shape. This result is stable for the three RootNet models, regardless of the input size, even if the numerical values slightly differ for each architecture. From the analysis of these curves, it is possible to define proper σ threshold values to analyze the results produced by RootNet. For example, fixing a desired precision and recall values of 0.95, it can be defined $\sigma_p \sim 0.6$ and $\sigma_r \sim 0.3$ to filter an image processed by RootNet with the following logic:

- If a pixel has a predicted outcome value greater than or equal to σ_p, it is labeled as *root*.
- If a pixel has a predicted outcome value less than or equal to σ_r, it is labeled as *background*.
- Otherwise, it is labeled as *unknown*.

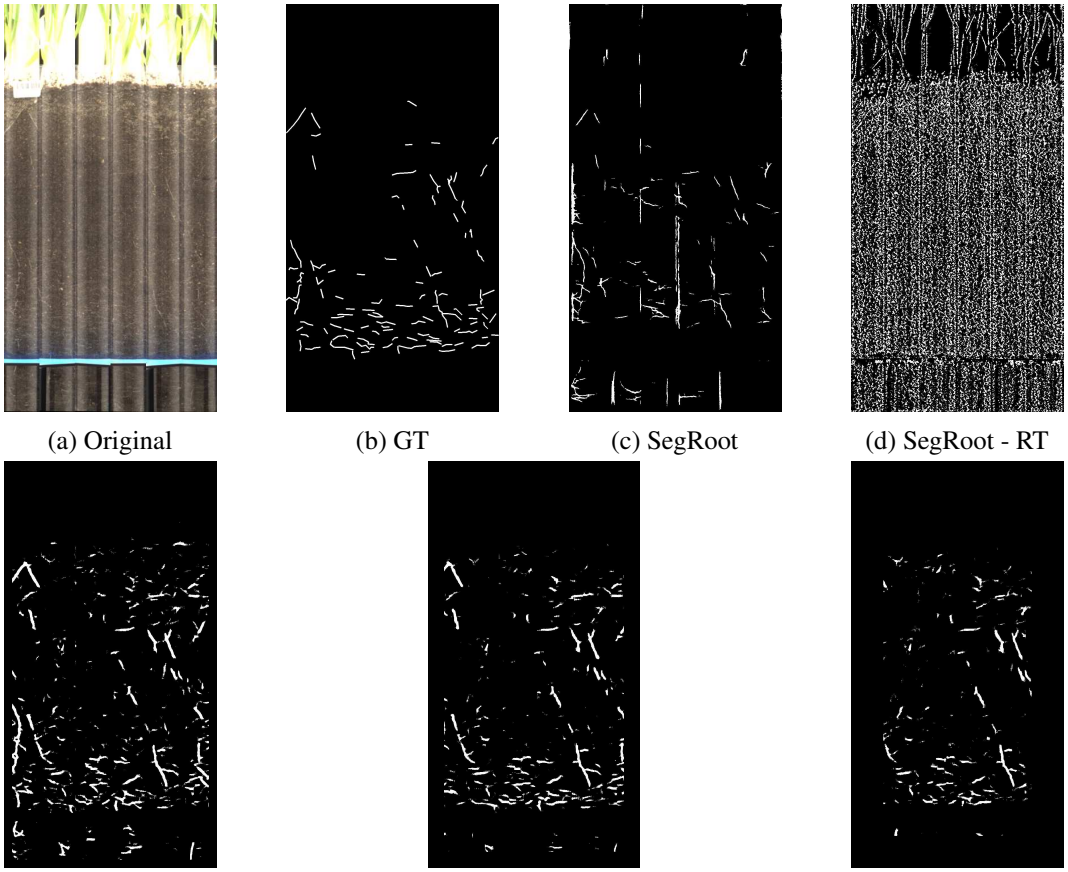
With reference to Figure 7.4, the same logic can be applied using a single threshold, obtained when $\sigma_p = \sigma_r = \arg\max_{\sigma}(F1) \sim 0.45$, as the higher values of the F1 score are achieved for a threshold value between 0.4 and 0.5. These threshold

values will be used in the next experiment to provide a qualitative evaluation of the images processed by RootNet, hence achieving a comparison with the SegRoot model on our dataset.

7.3.2 Comparison with SegRoot

This section contrasts the performance of RootNet and SegRoot. It is important to note that these two networks are based on different foundational architectures: SegRoot employs a U-shaped encoder/decoder framework, while RootNet utilizes a series of stacked Convolutional-ReLU-Max pooling layers capped with a binary classifier. Therefore, a direct quantitative comparison using traditional metrics (such as accuracy) might not be appropriate. As a result, a qualitative assessment is suggested, involving a comparison of the original image, the ground truth, and the outputs produced by both networks.

In Figure 7.5, a visual comparison of the results achieved by SegRoot and the three different versions of RootNet is shown.



(a) Original (b) GT (c) SegRoot (d) SegRoot - RT

(e) RootNet-65 (f) RootNet-129 (g) RootNet-257

Fig. 7.5 Results achieved on a sample image. From left to right: the original image (7.5a), the ground truth (7.5b) manually extracted by domain experts, the results achieved by SegNet with its original weights (7.5c) and after being retrained on our dataset (7.5d), and the results achieved by RootNet-65 (7.5e), RootNet-129 (7.5f), and RootNet-257 (7.5g), respectively.

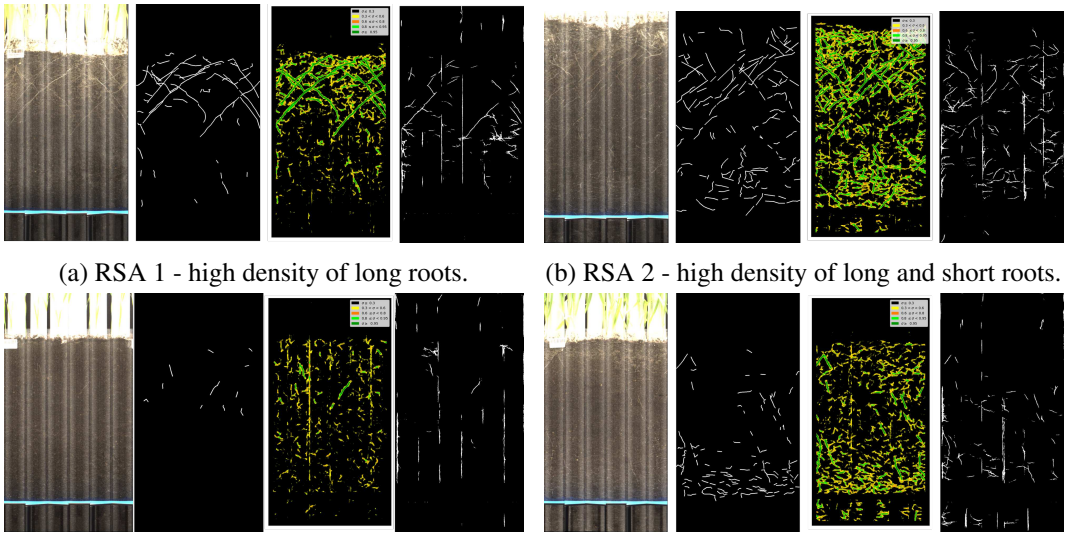
From Figure 7.5d, it can be seen that SegRoot provides the best results when used with the original weights, as retraining it on our dataset introduces a significant quantity of noise. However, by comparing the results achieved by SegRoot with the ground truth, it can be seen that it cannot capture the finer details, such as the smaller parts of the RSA. Furthermore, SegRoot misclassifies some of the artifacts introduced by the merging procedure applied on the original image (cfr.Section 7.2.1) as parts of the RSA. As for our architecture, the qualitative comparison shows that both RootNet-65 (Fig. 7.5e) and RootNet-129 (Fig. 7.5f) successfully capture fine-grained details about the RSA, with RootNet-129 achieving less noise in the bottom of the image, where no roots are available.

To further extend our comparison, let us qualitatively evaluate Figure 7.6, where the results achieved by SegRoot are compared with the ones achieved with RootNet-65 on four different RSAs, selected according to both the density and the length of available roots. Specifically:

- **RSA 1** (*top left*) has been selected since a high density of long roots is visible in the upper part of the image.
- **RSA 2** (*top right*) has been selected since there is a high density of both long and short roots over the whole image.
- **RSA 3** (*bottom left*) has been selected due to the low density of the visible short roots.
- **RSA 4** (*bottom right*) has been selected due to the high density of short roots in the bottom part of the image.

In each subfigure of Figure 7.6 is shown, from left to right, the original image, the ground truth, the results achieved using RootNet-65, and the results achieved by SegRoot with its original weights. Specifically, the results provided by RootNet-65 are described in terms of the values of *sigma* for each pixel. Hence:

- If the pixel is colored in dark green, the network has classified it as a root with a confidence score above 0.95.
- If the pixel is colored in green, the network has classified it as a root with a confidence score between 0.8 and 0.95.
- If the pixel is colored in orange, the network has classified it as a root with a confidence score between 0.6 and 0.8.
- If the pixel is colored in yellow, the network has classified it as a root with a confidence score between 0.3 and 0.6.
- If the pixel is colored in black, the network has classified it as a root with a confidence score below 0.3.



(a) RSA 1 - high density of long roots.

(b) RSA 2 - high density of long and short roots.

(c) RSA 3 - low density of roots.

(d) RSA 4 - High density of short roots.

Fig. 7.6 Qualitative comparison between SegRoot with original weights and RootNet-65. From left to right, respectively, the original image, the ground truth, RootNet-65 results, and finally SegRoot binary mask are reported.

In other words, according to the two-thresholds formulation described in Section 7.3.1, pixels colored in orange, green, and dark green can be considered roots with a high confidence level. On the other hand, pixels colored in black can be considered part of the background. Finally, pixels colored in yellow are labeled as "uncertain" and, as it can be seen, mostly belong to the zones relative to the artifacts introduced by the preprocessing on the images or to the zones surrounding the roots. This is also desirable, as it can highlight root parts that are effectively within the RSA but have not been labeled by the domain expert as too dim in their appearance on the image. As seen from the images, RootNet outperforms SegRoot in the cases shown in Figures 7.6a, 7.6b, and 7.6d, which account for dense zones of long and short roots, providing high reliability, especially by considering the two-thresholds formulation proposed. As for the case shown in Figure 7.6c, RootNet appears to be able to correctly characterize roots, which are also available in the ground truth; however, several points also appear with values of the confidence score above 0.6, which can be taken back in part to the artifacts within the image and in part to several dim structures within the RSA. Therefore, in this case, using the single-threshold formulation may be preferable.

7.3.3 Quantitative comparison

To further assess the performance of RootNet, a pixel-based quantitative comparison against SegRoot was proposed. Specifically, four metrics were used: precision, recall, F1 score and Hausdorff distance between the ground truth and a binary mask produced by each method. Results are shown in Table 7.3.

Table 7.3 Quantitative pixel-based comparison of RootNet against SegRoot.

Network	F1	P	R
SegRoot	11.58%	9.50%	20.42%
RootNet-65	17.56%	10.07%	**77.02%**
RootNet-129	**22.65%**	**13.77%**	67.78%
RootNet-257	21.40%	13.39%	60.62%

It is important to underline that, as already stated in Section 7.3.2, a direct comparison in terms of standard metrics among these models is not straightforward, as they are based on different considerations and working principles. In particular, the problem solved by RootNet is intrinsically formulated as a probability estimation, therefore the informative content output by the proposed method is not a simple binary mask. Moreover, to frame the problem and prepare the dataset, particular attention was payed to the ground truth labelling, privileging thin lines that certainly highlight a root in the images, due to the major voting procedure described before. As such, even if the quantitative comparison is based on a pixel-level evaluation of the results achieved using the networks in inference mode over 17 validation images, considering such binary masks and the ground truth labelled this way could lead to relatively low values of the F1 score. For this reason, to better evaluate the performance of the models in the most unbiased way, a computation of the Hausdorff distance between the ground truth and all the binary masks obtained by the networks was also provided. Table 7.3 shows that RootNet-129 outperforms the other models in terms of F1 score and precision, while RootNet-65 achieves the highest value for recall. Still, these values must be taken in the context of a pixel-based evaluation, which can be inherently biased by minimal offset errors in the prediction. In other words, a displacement of the prediction performed by the network of a negligible number of pixels, either vertically or horizontally, can significantly impact the values provided by the metrics. As such, the results were also validated considering the average prediction of a patch of 3×3 pixels. The results are shown in Table 7.4, and confirm the ones already achieved in Table 7.3.

Table 7.4 Quantitative comparison of RootNet against SegRoot over patches of 3×3 pixels.

Network	F1	P	R
SegRoot	11.66%	9.60%	20.32%
RootNet-65	18.39%	10.60%	**76.68%**
RootNet-129	**23.62%**	**14.48%**	67.70%
RootNet-257	22.41%	14.16%	60.40%

Finally, let us consider the border effect introduced by RootNet when used in inference. In fact, during the proposed tests, the model was used in inference without introducing any extra padding effect to avoid repetition bias. However, this imposes a tradeoff in that the outermost $\frac{N-1}{2}$ pixels will not be considered during the analysis, with N the patch size RootNet considers during training. Consequently, accounting for these border effects yields the results shown in Table 7.5.

Table 7.5 Quantitative comparison of RootNet considering the border effect.

Network	F1	P	R
RootNet-65	17.64%	10.06%	**79.33%**
RootNet-129	**23.07%**	**13.71%**	74.71%
RootNet-257	22.40%	13.39%	74.25%

Interestingly, when the border effect is considered, the precision is slightly affected, along with the overall F1 score, but the recall is noticeably improved. This is mainly related to the fact that the border pixels are not predicted as belonging to roots, hence the overall number of false negatives decreases, therefore improving the recall achievable by the network. Finally, in Table 7.6 the Hausdorff distance between the ground truth and the validation binary masks is reported, showing that the proposed approach is able to provide a root mask with a distance error of less than 2 pixels in the best case and less than 4 pixels in the worst case.

Table 7.6 Quantitative comparison of the Hausdorff distance between the ground truth and the binary masks computed by the network models.

Network	Hausdorff distance
SegRoot	24.61
RootNet-65	**1.78**
RootNet-129	3.27
RootNet-257	3.05

7.4 Summary

Computer vision and artificial intelligence are increasingly pivotal in plant phenotyping studies, enabling the analysis of vast sensor-gathered data. These studies assess complex plant traits in both aerial and underground parts, extracting valuable information on growth, development, tolerance, or resistance. The proposed method automatically and quantitatively evaluates plant traits in a non-destructive manner.

This chapter introduces a novel approach to Root System Architecture (RSA) segmentation, bypassing the need for a U-shaped network and instead utilizing binary classification via probability map estimation to distinguish root pixels from the background. This method has achieved optimal quantitative results in terms of accuracy and F1-score using compact Convolutional Neural Networks (CNNs) with just three stacked convolutional layers. The selected CNN model effectively demonstrates the end-to-end pipeline's efficiency, highlighting its reduced computational cost compared to more resource-intensive architectures.

This approach is particularly advantageous in scenarios with limited labeled data, as labeling RSAs is time-consuming and expensive for domain experts. By generating numerous patches from a relatively small number of images, the method provides a reliable tool for identifying complex RSAs. Furthermore, it can be easily scaled, with potential optimization to leverage GPUs for near-real-time performance on large datasets.

Chapter 8

Conclusions and future works

This thesis aimed to investigate and develop strategies to minimize the time and effort required for data management in the application of Deep Learning (DL) models and optimizing the architectures of these models to enhance performance, lower computational costs, and simplify their structure for high-throughput plant phenotyping.

This research aimed to systematically review hardware and software factors in high-performance plant phenotyping. It found that ground platforms are commonly used for their cost-effectiveness, simplicity, and compatibility with data collection. RGB cameras can capture both aerial and root system images, offering good resolution at lower costs. A key finding was the growing use of deep learning models in plant phenotyping, which excel at accurately identifying and measuring plant components like fruits, flowers, and roots. Deep neural networks outperform traditional methods, though attention is needed for custom model adaptation and generalization. YOLO (You Only Look Once) is an advanced object detection model designed to quickly and efficiently detect objects in images. The key feature of YOLO is that it processes an entire image at once, making it faster than traditional object detection models that analyze images in parts. Once YOLO is trained on labeled data—images that have already been annotated with the correct object classes and locations—it can predict the presence of specific traits or features in new, unseen images. The model works by dividing an image into a grid and simultaneously predicting bounding boxes and class probabilities for each grid cell. This allows YOLO to detect multiple objects in a single pass, providing real-time predictions. In applications

like agriculture and plant health research, YOLO's ability to analyze new images means it can monitor fields, crops, or plants autonomously without requiring prior knowledge of the specific images it encounters. For example, YOLO can detect signs of stress, disease, or other physiological changes in plants, even in environments that are constantly changing. This makes it a powerful tool for large-scale monitoring and real-time predictions, allowing for continuous and efficient data collection, which is crucial for tasks like precision agriculture, pest management, or crop health monitoring.

This thesis examined algorithms and efficient data management methods, focusing on CNN models for classifying plant traits (nodes, flowers, fruits) using both pre-trained and custom algorithms. A YOLOv5-based detector was proposed to identify tomatoes, flowers, and nodes, both individually and in combination. The research utilized RGB data with relatively low resolution, highlighting the innovative methodology's effectiveness with "limited resources." The study also explored the scalability of the proposed approach, demonstrating its potential for deployment with low-cost hardware, making it a feasible solution for large-scale plant phenotyping and stress monitoring.

To address data-related challenges, the next phase of this study focused on implementing solutions to improve data balance for more accurate diagnosis. A novel data-balancing approach was introduced, enhancing detection accuracy using attention mechanisms and transfer learning. The proposed solution involved implementing a YOLOv8 deep learning model, which effectively detects flowers, fruits, and nodes in tomato plants. Results showed that by incorporating attention mechanisms, along with a transfer-learning-like method to use best weights achieved on balanced data to evaluate imbalanced data, the accuracy in the detection of relevant phenotypical traits improved significantly. However, several limitations persist, primarily due to the need for incorporating semantic information into a framework that enables the network to differentiate between phenotypically related features, such as nodes on primary stems versus those on secondary stems. Additionally, the small size of fruits and flowers allows them to blend seamlessly with surrounding foliage. The minimal size contrast between the fruit and its background presents significant challenges for event detection in deep neural network models. With this in mind, the next phase of this thesis focuses on exploring and modifying the YOLOv8 architecture to address the complexities of datasets collected via a high-throughput phenotyping platform.

An innovative approach was developed to enhance the detection of small plant components, such as fruits and flowers, by modifying the YOLOv8 model architecture. The key innovation is the integration of a shallower P2 layer into the model's head, which enables the model to focus on low-level features essential for detecting small objects that may lack distinct characteristics and are easily obscured by the background. This modification improves the model's ability to capture crucial details, leading to better detection performance for small plant components. Additionally, the study introduces an integrated SO-YOLOv5 model, combining the YOLOv5 backbone with the YOLOv8 head. This hybrid innovative approach optimizes performance by leveraging the strengths of both architectures, resulting in a more efficient and effective model for detecting small plant traits. These architectural modifications enhance detection accuracy while minimizing computational costs and simplifying the model's structure.

Finally, this thesis presents a novel approach for identifying plant roots in root system architecture images using a convolutional neural network (CNN). The innovation of this approach lies in the development of a lightweight network architecture tailored for Root System Architecture (RSA) segmentation in resource-limited environments. Unlike existing solutions that rely on high-resolution cameras or specialized sensors, the proposed model effectively predicts "root pixels" in cluttered images captured with minimal camera specifications. This flexibility, combined with the model's ability to function without strict imaging conditions, makes it both adaptable and efficient. The simple architecture reduces computational demands while maintaining strong performance, offering a novel solution for RSA segmentation in environments with limited resources.

The limitations concerning this study and the direction for corresponding future studies are identified and discussed as follows:

- Further explore optimizations to enhance the model's efficiency and speed, with a specific focus on algorithms tailored to leverage the architecture of Nvidia Jetson or comparable specialized hardware platforms.
- Extend the research to encompass a wider array of plant species beyond those initially studied. This may involve compiling and annotating diverse datasets to effectively train and validate the model across various botanical families and plant types.

- Conduct comprehensive field testing to assess the model's robustness and accuracy in real-world agricultural environments. Create user-friendly interfaces or mobile applications that seamlessly integrate the model, enabling farmers and researchers to access real-time plant health assessments and recommendations directly from the field.
- Investigate the integration of additional data sources like multispectral or hyperspectral imaging, thermal imaging, and environmental sensors (e.g., humidity, temperature) to offer a comprehensive plant health assessment. Explore fusion techniques to effectively combine data from multiple modalities, enhancing the accuracy and reliability of the plant health assessment system.

In the concluding section of this thesis, the potential future applications of the research findings are explored. A primary challenge in future work will be the development of stable and robust models that can be trained on data derived from HTP platforms and effectively deployed in open-field environments. These models must not only exhibit the ability to adapt to the dynamic and unpredictable conditions of the field but also operate concurrently with real-time image acquisition and analysis. The successful integration of such models is essential for the practical application of these technologies in real-world scenarios. This research demonstrates the potential effectiveness of innovative approaches tailored to specific use cases, including the adaptation and refinement of existing models. Furthermore, future advancements will require addressing the scalability and generalization of these models to ensure their applicability across diverse agricultural contexts. This includes the integration of multisource data, such as environmental variables and sensor inputs, to enhance model accuracy and decision-making processes in real time. Additionally, enhancing the computational efficiency of these models will be critical to enable their deployment on low-latency, resource-constrained platforms, which are typical in field applications. Continued research will also focus on the seamless fusion of model predictions with autonomous systems, such as drones or robots, to facilitate proactive interventions in precision agriculture. Ultimately, this work lays the foundation for the next generation of intelligent, real-time systems capable of transforming agricultural practices through automation, data-driven insights, and sustainable decision-making.

In summary, this thesis outlines promising research directions in plant phenotyping. The ideas presented here provide a foundation for further exploration and innovation in the field. Future studies are expected to expand upon and enrich the findings discussed. With advancing technology and the introduction of new research methodologies, the possibilities for future research are vast. The field of plant phenotyping is evolving rapidly, offering exciting opportunities for future breakthroughs and discoveries.

References

[1] Abade, A., Ferreira, P. A., and de Barros Vidal, F. (2021). Plant diseases recognition on images using convolutional neural networks: A systematic review. *Computers and Electronics in Agriculture*, 185:106125.

[2] Abu-Elsaoud, A. M. and Abdel-Azeem, A. M. (2020). Light, electromagnetic spectrum, and photostimulation of microorganisms with special reference to chaetomium. *Recent Developments on Genus Chaetomium*, pages 377–393.

[3] Ahmed, M. R., Yasmin, J., Park, E., Kim, G., Kim, M. S., Wakholi, C., Mo, C., and Cho, B.-K. (2020). Classification of watermelon seeds using morphological patterns of x-ray imaging: A comparison of conventional machine learning and deep learning. *Sensors*, 20(23):6753.

[4] Albelwi, S. and Mahmood, A. (2017). A framework for designing the architectures of deep convolutional neural networks. *Entropy*, 19(6):242.

[5] Altieri, M. A. and Nicholls, C. I. (2017). The adaptation and mitigation potential of traditional agriculture in a changing climate. *Climatic change*, 140:33–45.

[6] Araus, J. L. and Cairns, J. E. (2014). Field high-throughput phenotyping: the new crop breeding frontier. *Trends in plant science*, 19(1):52–61.

[7] Armengaud, P. (2009). Ez-rhizo software: the gateway to root architecture analysis. *Plant signaling & behavior*, 4(2):139–141.

[8] Atefi, A., Ge, Y., Pitla, S., and Schnable, J. (2021). Robotic technologies for high-throughput plant phenotyping: Contemporary reviews and future perspectives. *Frontiers in plant science*, 12:611940.

[9] Atkinson, J. A., Pound, M. P., Bennett, M. J., and Wells, D. M. (2019). Uncovering the hidden half of plants using new advances in root phenotyping. *Current opinion in biotechnology*, 55:1–8.

[10] Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495.

[11] Banerjee, B. P., Sharma, V., Spangenberg, G., and Kant, S. (2021). Machine learning regression analysis for estimation of crop emergence using multispectral uav imagery. *Remote Sensing*, 13(15):2918.

[12] Bareth, G., Aasen, H., Bendig, J., et al. (2016). Low-weight and uav-based hyperspectral full-frame cameras for monitoring crops: spectral comparison with portable spectroradiometer measurements. *Photogrammetric Engineering & Remote Sensing*, 82(8):663–672.

[13] Bauer, F. M., Lärm, L., Morandage, S., Lobet, G., Vanderborght, J., Vereecken, H., and Schnepf, A. (2022). Development and validation of a deep learning-based automated minirhizotron image analysis pipeline. *Plant Phenomics*.

[14] Berger, K., Atkinson, P. M., and Dawson, T. P. (2020). Remote sensing for high-throughput plant phenotyping: the challenges of scale. *Frontiers in Plant Science*, 11:456.

[15] Biber, P., Weiss, U., Dorna, M., and Albert, A. (2012). Navigation system of the autonomous agricultural robot bonirob. In *Workshop on Agricultural Robotics: Enabling Safe, Efficient, and Affordable Robots for Food Production (Collocated with IROS 2012), Vilamoura, Portugal*.

[16] Boogaard, F. P., Rongen, K. S., and Kootstra, G. W. (2020). Robust node detection and tracking in fruit-vegetable crops using deep learning and multi-view imaging. *Biosystems Engineering*, 192:117–132.

[17] Borianne, P., Subsol, G., Fallavier, F., Dardou, A., and Audebert, A. (2018). Gt-roots: an integrated software for automated root system measurement from high-throughput phenotyping platform images. *Computers and electronics in agriculture*, 150:328–342.

[18] Borra Serrano, I. (2020). High-throughput field phenotyping using a drone with rgb imagery. exploiting the spectral, spatial and temporal dimensions.

[19] Boureau, Y., Ponce, J., and LeCun, Y. (2010). A theoretical analysis of feature pooling in visual recognition. *Proceedings of the International Conference on Machine Learning (ICML)*, pages 111–118.

[20] Brichet, N., Fournier, C., Turc, O., Strauss, O., Artzet, S., Pradal, C., Welcker, C., Tardieu, F., and Cabrera-Bosquet, L. (2017). A robot-assisted imaging pipeline for tracking the growths of maize ear and silks in a high-throughput phenotyping platform. *Plant Methods*, 13:1–12.

[21] Bucksch, A., Burridge, J., York, L. M., Das, A., Nord, E., Weitz, J. S., and Lynch, J. P. (2014). Image-based high-throughput field phenotyping of crop roots. *Plant Physiology*, 166(2):470–486.

[22] Cai, Z. and Vasconcelos, N. (2018). Cascade r-cnn: Delving into high-quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162.

[23] Calvo, P., Nelson, L., and Kloepper, J. W. (2014). Agricultural uses of plant biostimulants. *Plant and soil*, 383:3–41.

[24] Cardellicchio, A., Solimani, F., Dimauro, G., Petrozza, A., Summerer, S., Cellini, F., and Renò, V. (2023). Detection of tomato plant phenotyping traits using yolov5-based single stage detectors. *Computers and Electronics in Agriculture*, 207:107757.

[25] Cardellicchio, A., Solimani, F., Dimauro, G., Summerer, S., and Renò, V. (2024). Patch-based probabilistic identification of plant roots using convolutional neural networks. *Pattern Recognition Letters*, 183:125–132.

[26] Carvalho, P., Lourenço, N., Assunção, F., and Machado, P. (2020). Autolr: An evolutionary approach to learning rate policies. In *Proceedings of the 2020 genetic and evolutionary computation conference*, pages 672–680.

[27] Chai, J. J., Xu, J.-L., and O'Sullivan, C. (2023). Real-time detection of strawberry ripeness using augmented reality and deep learning. *Sensors*, 23(17):7639.

[28] Chaudhury, A., Ward, C., Talasaz, A., Ivanov, A. G., Brophy, M., Grodzinski, B., Hüner, N. P., Patel, R. V., and Barron, J. L. (2018). Machine vision system for 3d plant phenotyping. *IEEE/ACM transactions on computational biology and bioinformatics*, 16(6):2009–2022.

[29] Chavarría-Krauser, A., Nagel, K. A., Palme, K., Schurr, U., Walter, A., and Scharr, H. (2008). Spatio-temporal quantification of differential growth processes in root growth zones based on a novel combination of image sequence processing and refined concepts describing curvature production. *New Phytologist*, 177(3):811–821.

[30] Chawade, A., van Ham, J., Blomquist, H., Bagge, O., Alexandersson, E., and Ortiz, R. (2019). High-throughput field-phenotyping tools for plant breeding and precision agriculture. *Agronomy*, 9(5):258.

[31] Chen, D., Neumann, K., Friedel, S., Kilian, B., Chen, M., Altmann, T., and Klukas, C. (2014). Dissecting the phenotypic components of crop plant growth and drought responses based on high-throughput image analysis. *The plant cell*, 26(12):4636–4655.

[32] Chen, L. et al. (2022). Integrating deep learning with advanced imaging for plant stress analysis. *Frontiers in Plant Science*.

[33] Chen, S., Liao, Y., Lin, F., and Huang, B. (2023). An improved lightweight yolov5 algorithm for detecting strawberry diseases. *IEEE Access*.

[34] Chen, Z., Wang, J., Wang, T., Song, Z., Li, Y., Huang, Y., Wang, L., and Jin, J. (2021). Automated in-field leaf-level hyperspectral imaging of corn plants using a cartesian robotic platform. *Computers and Electronics in Agriculture*, 183:105996.

[35] Ciampitti, I. A., Murrell, S. T., Camberato, J. J., Tuinstra, M., Xia, Y., Friedemann, P., and Vyn, T. J. (2013). Physiological dynamics of maize nitrogen uptake and partitioning in response to plant density and nitrogen stress factors: Ii. reproductive phase. *Crop Science*, 53(6):2588–2602.

[36] Cobb, J. N., DeClerck, G., Greenberg, A., Clark, R., and McCouch, S. (2013). Next-generation phenotyping: requirements and strategies for enhancing our understanding of genotype–phenotype relationships and its relevance to crop improvement. *Theoretical and Applied Genetics*, 126:867–887.

[37] Colla, G., Rouphael, Y., et al. (2015). Biostimulants in horticulture. *Scientia Horticulturae*, 196:1–134.

[38] Costa, J. M., Grant, O. M., and Chaves, M. M. (2013). Thermography to explore plant–environment interactions. *Journal of experimental botany*, 64(13):3937–3949.

[39] Cui, M., Jiang, Q., Li, N., and Xue, X. (2021). Research on strawberry maturity detection technology based on improved yolov4. In *Journal of Physics: Conference Series*, volume 2138, page 012012. IOP Publishing.

[40] Czedik-Eysenberg, A., Seitner, S., Güldener, U., Koemeda, S., Jez, J., Colombini, M., and Djamei, A. (2018). The 'phenobox', a flexible, automated, open-source plant phenotyping solution. *New Phytologist*, 219(2):808–823.

[41] Da Silva, C. B., Bianchini, V. d. J. M., de Medeiros, A. D., de Moraes, M. H. D., Marassi, A. G., and Tannus, A. (2021). A novel approach for jatropha curcas seed health analysis based on multispectral and resonance imaging techniques. *Industrial Crops and Products*, 161:113186.

[42] Danzi, D., Briglia, N., Petrozza, A., Summerer, S., Povero, G., Stivaletta, A., Cellini, F., Pignone, D., De Paola, D., and Janni, M. (2019). Can high throughput phenotyping help food security in the mediterranean area? *Frontiers in Plant Science*, 10:15.

[43] Das Choudhury, S., Guha, S., Das, A., Das, A. K., Samal, A., and Awada, T. (2022). Flowerphenonet: Automated flower detection from multi-view image sequences using deep neural networks for temporal plant phenotyping analysis. *Remote Sensing*, 14(24):6252.

[44] Daviet, B., Fernandez, R., Cabrera-Bosquet, L., Pradal, C., and Fournier, C. (2022). Phenotrack3d: an automatic high-throughput phenotyping pipeline to track maize organs over time. *Plant Methods*, 18(1):130.

[45] Diba, A., Sharma, V., Pazandeh, A., Pirsiavash, H., and Van Gool, L. (2017). Weakly supervised cascaded convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 914–922. IEEE.

[46] Ding, J., Ren, X., Luo, R., and Sun, X. (2019). An adaptive and momental bound method for stochastic learning. *arXiv preprint arXiv:1910.12249*.

[47] Downie, H., Holden, N., Otten, W., Spiers, A. J., Valentine, T. A., and Dupuy, L. X. (2012). Transparent soil for imaging the rhizosphere. *PLoS ONE*, 7(9):e44276.

[48] Du, J., Lu, X., Fan, J., Qin, Y., Yang, X., and Guo, X. (2020). Image-based high-throughput detection and phenotype evaluation method for multiple lettuce varieties. *Frontiers in Plant Science*, 11:563386.

[49] Du, X., Cheng, H., Ma, Z., Lu, W., Wang, M., Meng, Z., Jiang, C., and Hong, F. (2023). Dsw-yolo: A detection method for ground-planted strawberry fruits under different occlusion levels. *Computers and Electronics in Agriculture*, 214:108304.

[50] ElManawy, A. I., Sun, D., Abdalla, A., Zhu, Y., and Cen, H. (2022). Hsi-pp: A flexible open-source software for hyperspectral imaging-based plant phenotyping. *Computers and Electronics in Agriculture*, 200:107248.

[51] Fahlgren, N., Gehan, M. A., and Baxter, I. (2015). Lights, camera, action: high-throughput plant phenotyping is ready for a close-up. *Current opinion in plant biology*, 24:93–99.

[52] Falk, K. G., Jubery, T. Z., Mirnezami, S. V., Parmley, K. A., Sarkar, S., Singh, A., Ganapathysubramanian, B., and Singh, A. K. (2020). Computer vision and machine learning enabled soybean root phenotyping pipeline. *Plant methods*, 16:1–19.

[53] Fan, J., Zhang, Y., Wen, W., Gu, S., Lu, X., and Guo, X. (2021). The future of internet of things in agriculture: Plant high-throughput phenotypic platform. *Journal of Cleaner Production*, 280:123651.

[54] Fan, S., Liang, X., Huang, W., Zhang, V. J., Pang, Q., He, X., Li, L., and Zhang, C. (2022). Real-time defects detection for apple sorting using nir cameras with pruning-based yolov4 network. *Computers and Electronics in Agriculture*, 193:106715.

[55] Feng, L., Chen, S., Zhang, C., Zhang, Y., and He, Y. (2021). A comprehensive review on recent applications of unmanned aerial vehicle remote sensing with various sensors for high-throughput plant phenotyping. *Computers and electronics in agriculture*, 182:106033.

[56] Fiorani, F. and Schurr, U. (2013). Future scenarios for plant phenotyping. *Annual review of plant biology*, 64(1):267–291.

[57] Flavel, R. J., Guppy, C. N., Tighe, M., Watt, M., McNeill, A., and Young, I. M. (2012). Non-destructive quantification of cereal roots in soil using high-resolution x-ray tomography. *Journal of Experimental Botany*, 63(7):2503–2511.

[58] Fu, X. and Jiang, D. (2022). High-throughput phenotyping: The latest research tool for sustainable crop production under global climate change scenarios. In *Sustainable Crop Productivity and Quality Under Climate Change*, pages 313–381. Elsevier.

[59] Fukushima, K. (1988). Neocognitron: A hierarchical neural network capable of visual pattern recognition. *Neural Networks*, 1:119–130.

[60] Gage, J. L., Richards, E., Lepak, N., Kaczmar, N., Soman, C., Chowdhary, G., Gore, M. A., and Buckler, E. S. (2019). In-field whole-plant maize architecture characterized by subcanopy rovers and latent space phenotyping. *The Plant Phenome Journal*, 2(1):1–11.

[61] Gai, R., Chen, N., and Yuan, H. (2023). A detection algorithm for cherry fruits based on the improved yolo-v4 model. *Neural Computing and Applications*, 35(19):13895–13906.

[62] Galkovskyi, T., Mileyko, Y., Bucksch, A., Moore, B., Symonova, O., Price, C. A., Topp, C. N., Iyer-Pascuzzi, A. S., Zurek, P. R., Fang, S., et al. (2012). Gia roots: software for the high throughput analysis of plant root system architecture. *BMC plant biology*, 12:1–12.

[63] Gao, Z., Luo, Z., Zhang, W., Lv, Z., and Xu, Y. (2020). Deep learning application in plant stress imaging: A review. agriengineering, 2 (3): 430-446.

[64] Gebbers, R. and Adamchuk, V. I. (2010). Precision agriculture and food security. *Science*, 327(5967):828–831.

[65] Gerhards, M., Schlerf, M., Mallick, K., and Udelhoven, T. (2019). Challenges and future perspectives of multi-/hyperspectral thermal infrared remote sensing for crop water-stress detection: A review. *Remote Sensing*, 11(10):1240.

[66] Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448.

[67] Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587.

[68] Gong, L., Du, X., Zhu, K., Lin, C., Lin, K., Wang, T., Lou, Q., Yuan, Z., Huang, G., and Liu, C. (2021). Pixel level segmentation of early-stage in-bag rice root for its architecture analysis. *Computers and Electronics in Agriculture*, 186:106197.

[69] Gracia-Romero, A., Kefauver, S. C., Vergara-Díaz, O., Zaman-Allah, M. A., Prasanna, B. M., Cairns, J. E., and Araus, J. L. (2017). Comparative performance of ground vs. aerially assessed rgb and multispectral indices for early-growth evaluation of maize performance under phosphorus fertilization. *Frontiers in Plant Science*, 8:309121.

[70] Granier, C. and Vile, D. (2014). Phenotyping and beyond: modelling the relationships between traits. *Current opinion in plant biology*, 18:96–102.

[71] Gray, G. R., Hope, B. J., Qin, X., Taylor, B. G., and Whitehead, C. L. (2003). The characterization of photoinhibition and recovery during cold acclimation in arabidopsis thaliana using chlorophyll fluorescence imaging. *Physiologia Plantarum*, 119(3):365–375.

[72] Grimstad, L. and From, P. J. (2017). Thorvald ii-a modular and reconfigurable agricultural robot. *IFAC-PapersOnLine*, 50(1):4588–4593.

[73] Guangyu, C., Taihui, Z., Jian, M., Yi, S., and Xin, C. (2012). Changes in spectral reflectance of vegetation in response to specific nutrient supply. In *Advances in Electric and Electronics*, pages 671–675. Springer.

[74] Guo, X., Qiu, Y., Nettleton, D., Yeh, C.-T., Zheng, Z., Hey, S., and Schnable, P. S. (2021). Kat4ia: K-means assisted training for image analysis of field-grown plant phenotypes. *Plant Phenomics*.

[75] Häni, N., Roy, P., and Isler, V. (2020). A comparative study of fruit detection and counting methods for yield mapping in apple orchards. *Journal of Field Robotics*, 37(2):263–282.

[76] He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017a). Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2961–2969.

[77] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

[78] He, K., Zhang, X., Ren, S., and Sun, J. (2017b). Deep residual learning for image recognition. corr 2015; abs/1512.03385. *arXiv preprint arXiv:1512.03385*.

[79] He, Z., Khana, S. R., Zhang, X., Karkee, M., and Zhang, Q. (2023). Real-time strawberry detection based on improved yolov5s architecture for robotic harvesting in open-field environment. *arXiv preprint arXiv:2308.03998*.

[80] Helliwell, J., Sturrock, C. J., Mairhofer, S., Craigon, J., Ashton, R., Miller, A., Whalley, W., and Mooney, S. J. (2017). The emergent rhizosphere: imaging the development of the porous architecture at the root-soil interface. *Scientific reports*, 7(1):14875.

[81] Hernández-Sánchez, N., Hills, B., Barreiro, P., and Marigheto, N. (2007). An nmr study on internal browning in pears. *Postharvest Biology and Technology*, 44(3):260–270.

[82] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

[83] Hu, J., Shen, L., and Sun, G. (2018). Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141.

[84] Hu, P., Chapman, S. C., and Zheng, B. (2021). Coupling of machine learning methods to improve estimation of ground coverage from unmanned aerial vehicle (uav) imagery for high-throughput phenotyping of crops. *Functional Plant Biology*, 48(8):766–779.

[85] Hu, T. and Zhang, J. (2023). Highly overlapping strawberry leaf detection based on yolov5s-mbls deep learning method. In *2023 3rd International Conference on Neural Networks, Information and Communication Engineering (NNICE)*, pages 145–148. IEEE.

[86] Jiang, Y. and Li, C. (2020). Convolutional neural networks for image-based high-throughput plant phenotyping: a review. *Plant Phenomics*.

[87] Jiang, Y., Li, C., Xu, R., Sun, S., Robertson, J. S., and Paterson, A. H. (2020). Deepflower: a deep learning-based approach to characterize flowering patterns of cotton plants in the field. *Plant methods*, 16:1–17.

[88] Jocher, G. and Chaurasia, A. (2023). Ultralytics. *Accessed on Jun*, 9.

[89] Jubery, T. Z., Carley, C. N., Singh, A., Sarkar, S., Ganapathysubramanian, B., and Singh, A. K. (2021). Using machine learning to develop a fully automated soybean nodule acquisition pipeline (snap). *Plant Phenomics*.

[90] Kamilaris, A. and Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 147:70–90.

[91] Köckenberger, W., De Panfilis, C., Santoro, D., Dahiya, P., and Rawsthorne, S. (2004). High-resolution nmr microscopy of plants and fungi. *Journal of Microscopy*, 214(2):182–189.

[92] Koh, J. C., Spangenberg, G., and Kant, S. (2021). Automated machine learning for high-throughput image-based plant phenotyping. *Remote Sensing*, 13(5):858.

[93] Kotu, V. and Deshpande, B. (2019). Data science process. *Data science*, pages 19–37.

[94] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.

[95] Kumar, A., Singh, A., Ganapathysubramanian, B., and Sarkar, S. (2021). High-throughput plant phenotyping: A key enabler for crop improvement. *Journal of Experimental Botany*, 72(2):345–362.

[96] Kumar, P., Huang, C., Cai, J., and Miklavcic, S. J. (2014). Root phenotyping by root tip detection and classification through statistical learning. *Plant and soil*, 380:193–209.

[97] Kunstner, F., Chen, J., Lavington, J. W., and Schmidt, M. (2023). Noise is not the main factor behind the gap between sgd and adam on transformers, but sign descent might be. *arXiv preprint arXiv:2304.13960*.

[98] Lawal, M. O. (2021a). Tomato detection based on modified yolov3 framework. *Scientific Reports*, 11(1):1–11.

[99] Lawal, O. M. (2021b). Development of tomato detection model for the robotic platform using deep learning. *Multimedia Tools and Applications*, 80(17):26751–26772.

[100] Le Bot, J., Serra, V., Fabre, J., Draye, X., Adamowicz, S., and Pagès, L. (2010). Dart: a software to analyse root system architecture and development from captured images. *Plant and Soil*, 326:261–273.

[101] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

[102] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

[103] Leiva, F. (2023). Developing affordable high-throughput plant phenotyping methods for breeding of cereals and tuber crops. *Acta Universitatis Agriculturae Sueciae*, (2023: 25).

[104] Li, D., Li, C., Yao, Y., Li, M., and Liu, L. (2020a). Modern imaging techniques in plant nutrition analysis: A review. *Computers and Electronics in Agriculture*, 174:105459.

[105] Li, J., Zhu, Z., Liu, H., Su, Y., and Deng, L. (2023a). Strawberry r-cnn: Recognition and counting model of strawberry based on improved faster r-cnn. *Ecological Informatics*, 77:102210.

[106] Li, L., Zhang, Q., and Huang, D. (2014). A review of imaging techniques for plant phenotyping. *Sensors*, 14(11):20078–20111.

[107] Li, L., Zhang, Q., and Huang, D. (2020b). Noninvasive techniques for plant phenotyping: A review. *Computers and Electronics in Agriculture*, 176:105672.

[108] Li, R., Ji, Z., Hu, S., Huang, X., Yang, J., and Li, W. (2023b). Tomato maturity recognition model based on improved yolov5 in greenhouse. *Agronomy*, 13(2):603.

[109] Li, X. et al. (2021). Deep learning enables rapid processing of extensive datasets for large-scale phenotyping in modern agriculture. *Plant Phenomics*.

[110] Lieder, I., Resheff, Y. S., and T.H. (2017). Learning tensorflow. In *Learning TensorFlow*, chapter 4.

[111] Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2117–2125.

[112] Lin, Y., Cai, R., Lin, P., and Cheng, S. (2022). A detection approach for bundled log ends using k-median clustering and improved yolov4-tiny network. *Computers and Electronics in Agriculture*, 194:106700.

[113] Litvin, A. G., van Iersel, M. W., and Malladi, A. (2016). Drought stress reduces stem elongation and alters gibberellin-related gene expression during vegetative growth of tomato. *Journal of the American Society for Horticultural Science*, 141(6):591–597.

[114] Liu, G., Nouaze, J. C., Touko Mbouembe, P. L., and Kim, J. H. (2020). Yolo-tomato: A robust algorithm for tomato detection based on yolov3. *Sensors*, 20(7):2145.

[115] Liu, J. and Wang, X. (2020). Tomato diseases and pests detection based on improved yolo v3 convolutional neural network. *Frontiers in plant science*, 11:521544.

[116] Llugsi, R., El Yacoubi, S., Fontaine, A., and Lupera, P. (2021). Comparison between adam, adamax and adam w optimizers to implement a weather forecast based on neural networks for the andean city of quito. In *2021 IEEE Fifth Ecuador Technical Chapters Meeting (ETCM)*, pages 1–6. IEEE.

[117] Lobet, G., Pagès, L., and Draye, X. (2011). A novel image-analysis toolbox enabling quantitative analysis of root system architecture. *Plant physiology*, 157(1):29–39.

[118] Lobet, G., Pagès, L., and Draye, X. (2013). A novel image-analysis toolbox enabling quantitative analysis of root system architecture. *Plant Physiology*, 162(4):1802–1813.

[119] Lorence, A. and Jimenez, K. M. (2022). *High-Throughput Plant Phenotyping: Methods and Protocols*, volume 2539. Springer Nature.

[120] Lu, H. and Cao, Z. (2020). Tasselnetv2+: A fast implementation for high-throughput plant counting from high-resolution rgb imagery. *Frontiers in plant science*, 11:541960.

[121] Lu, H., Tang, L., Whitham, S. A., and Mei, Y. (2017). A robotic platform for corn seedling morphological traits characterization. *Sensors*, 17(9):2082.

[122] Lu, Z., Miao, J., Dong, J., Zhu, S., Wu, P., Wang, X., and Feng, J. (2023). Automatic multilabel classification of multiple fundus diseases based on convolutional neural network with squeeze-and-excitation attention. *Translational Vision Science & Technology*, 12(1):22–22.

[123] Lube, V., Noyan, M. A., Przybysz, A., Salama, K., and Blilou, I. (2022). Multiplexlab: A high-throughput portable live-imaging root phenotyping platform using deep learning and computer vision. *Plant Methods*, 18(1):38.

[124] Luo, L., Xiong, Y., Liu, Y., and Sun, X. (2019). Adaptive gradient methods with dynamic bound of learning rate. *arXiv preprint arXiv:1902.09843*.

[125] Magalhães, S. A., Castro, L., Moreira, G., Dos Santos, F. N., Cunha, M., Dias, J., and Moreira, A. P. (2021). Evaluating the single-shot multibox detector and yolo deep learning models for the detection of tomatoes in a greenhouse. *Sensors*, 21(10):3569.

[126] Mahlein, A.-K., Oerke, E.-C., Steiner, U., and Dehne, H.-W. (2012a). Plant disease detection by hyperspectral imaging: A review. *Crop Protection*, 34:1–11.

[127] Mahlein, A.-K., Steiner, U., Hillnhütter, C., Dehne, H.-W., and Oerke, E.-C. (2012b). Hyperspectral imaging for small-scale analysis of symptoms caused by different sugar beet diseases. *Plant methods*, 8:1–13.

[128] Maji, A. K., Marwaha, S., Kumar, S., Arora, A., Chinnusamy, V., and Islam, S. (2022). Slypnet: Spikelet-based yield prediction of wheat using advanced plant phenotyping and computer vision techniques. *Frontiers in Plant Science*, 13:889853.

[129] Mbouembe, P. L. T., Liu, G., Sikati, J., Kim, S. C., and Kim, J. H. (2023). An efficient tomato-detection method based on improved yolov4-tiny model in complex environment. *Frontiers in Plant Science*, 14:1150958.

[130] McBratney, A., Whelan, B., Ancev, T., and Bouma, J. (2005). Future directions of precision agriculture. *Precision agriculture*, 6:7–23.

[131] Meraj, T., Sharif, M., Raza, M., Alabrah, A., Kadry, S., and Gandomi, A. (2024). Computer vision-based plants phenotyping: A comprehensive survey. *iScience*, 27(1).

[132] Mesa, T., Polo, J., Arabia, A., Caselles, V., and Munné-Bosch, S. (2022). Differential physiological response to heat and cold stress of tomato plants and its implication on fruit quality. *Journal of Plant Physiology*, 268:153581.

[133] Milella, A., Marani, R., Petitti, A., and Reina, G. (2019). In-field high throughput grapevine phenotyping with a consumer-grade depth camera. *Computers and electronics in agriculture*, 156:293–306.

[134] Milletari, F., Navab, N., and Ahmadi, S.-A. (2016). V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pages 565–571. Ieee.

[135] Minervini, M., Giuffrida, M. V., Perata, P., and Tsaftaris, S. A. (2017). Phenotiki: An open software and hardware platform for affordable and easy image-based phenotyping of rosette-shaped plants. *The Plant Journal*, 90(1):204–216.

[136] Minervini, M., Scharr, H., and Tsaftaris, S. A. (2015). Image analysis: The new bottleneck in plant phenotyping. *IEEE Signal Processing Magazine*, 32(4):126–131.

[137] Moghimi, A., Yang, C., and Anderson, J. A. (2020). Aerial hyperspectral imagery and deep neural networks for high-throughput yield phenotyping in wheat. *Computers and Electronics in Agriculture*, 172:105299.

[138] Mohammed, G. H., Noland, T. L., Irving, P., Sampson, P., Zarco-Tejada, P. J., and Miller, J. R. (2000). *Natural and stress-induced effects on leaf spectral reflectance in Ontario species*. Ontario Ministry of Natural Resources.

[139] Moher, D., Liberati, A., Tetzlaff, J., Altman, D. G., and PRISMA Group*, t. (2009). Preferred reporting items for systematic reviews and meta-analyses: the prisma statement. *Annals of internal medicine*, 151(4):264–269.

[140] Möller, B., Chen, H., Schmidt, T., Zieschank, A., Patzak, R., Türke, M., Weigelt, A., and Posch, S. (2019). rhizotrak: a flexible open source fiji plugin for user-friendly manual annotation of time-series images from minirhizotrons. *Plant and Soil*, 444:519–534.

[141] Mooney, S. J., Pridmore, T. P., Helliwell, J., and Bennett, M. J. (2012a). Developing x-ray computed tomography to non-invasively image 3-d root systems architecture in soil. *Plant and soil*, 352:1–22.

[142] Mooney, S. J., Pridmore, T. P., Helliwell, J., and Bennett, M. J. (2012b). Root phenotyping: An integrated approach to exploiting the root system for crop improvement. *Functional Plant Biology*, 39(11):923–937.

[143] Mosley, L., Pham, H., Bansal, Y., and Hare, E. (2020). Image-based sorghum head counting when you only look once. *arXiv preprint arXiv:2009.11929*.

[144] Mu, Y., Chen, T.-S., Ninomiya, S., and Guo, W. (2020). Intact detection of highly occluded immature tomatoes on plants using deep learning techniques. *Sensors*, 20(10):2984.

[145] Mueller-Sim, T., Jenkins, M., Abel, J., and Kantor, G. (2017). The robotanist: A ground-based agricultural robot for high-throughput crop phenotyping. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 3634–3639. IEEE.

[146] Murman, J. N. (2019). Flex-ro: a robotic high throughput field phenotyping system.

[147] Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., and Sutskever, I. (2021). Deep double descent: Where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124003.

[148] Narisetti, N., Henke, M., Seiler, C., Shi, R., Junker, A., Altmann, T., and Gladilin, E. (2019). Semi-automated root image analysis (saria). *Scientific Reports*, 9(1):19674.

[149] Näsi, R., Viljanen, N., Kaivosoja, J., Alhonoja, K., Hakala, T., Markelin, L., and Honkavaara, E. (2018). Estimating biomass and nitrogen amount of barley and grass using uav and aircraft based spectral and photogrammetric 3d features. *Remote Sensing*, 10(7):1082.

[150] Nations, U. (2019). World population prospects 2019. *Vol (ST/ESA/SE. A/424) Department of Economic and Social Affairs: Population Division*.

[151] Neilson, E. H., Edwards, A. M., Blomstedt, C., Berger, B., Møller, B. L., and Gleadow, R. M. (2015). Utilization of a high-throughput shoot imaging system to examine the dynamic phenotypic responses of a c4 cereal crop plant to nitrogen and water deficiency over time. *Journal of experimental botany*, 66(7):1817–1832.

[152] Panguluri, S. K. and Kumar, A. A. (2016). *Phenotyping for plant breeding*. Springer.

[153] Panthee, D. R., Kressin, J. P., and Piotrowski, A. (2018). Heritability of flower number and fruit set under heat stress in tomato. *HortScience*, 53(9):1294–1299.

[154] Pérez-Borrero, I., Marín-Santos, D., Gegundez-Arias, M. E., and Cortés-Ancos, E. (2020). A fast and accurate deep learning method for strawberry instance segmentation. *Computers and Electronics in Agriculture*, 178:105736.

[155] Perret, J., Al-Belushi, M., and Deadman, M. (2007). Non-destructive visualization and quantification of roots using computed tomography. *Soil Biology and Biochemistry*, 39(2):391–399.

[156] Petti, D. and Li, C. (2022). Weakly-supervised learning to automatically count cotton flowers from aerial imagery. *Computers and electronics in agriculture*, 194:106734.

[157] Phillips, R. L. (2010). Mobilizing science to break yield barriers. *Crop Science*, 50:S–99.

[158] Pieruschka, R. and Poorter, H. (2012). Phenotyping plants: genes, phenes and machines. *Functional Plant Biology*, 39(11):813–820.

[159] Planchamp, C., Balmer, D., Hund, A., and Mauch-Mani, B. (2013). A soil-free root observation system for the study of root-microorganism interactions in maize. *Plant and soil*, 367:605–614.

[160] Pound, M. P., Atkinson, J. A., Townsend, A. J., Wilson, M. H., Griffiths, M., Jackson, A. S., Bulat, A., Tzimiropoulos, G., Wells, D. M., Murchie, E. H., et al. (2017). Deep machine learning provides state-of-the-art performance in image-based plant phenotyping. *GigaScience*, 6(10):1–10.

[161] Pound, M. P., French, A. P., Atkinson, J. A., Wells, D. M., Bennett, M. J., and Pridmore, T. (2013). Rootnav: navigating images of complex root architectures. *Plant physiology*, 162(4):1802–1814.

[162] Pranga, J., Borra-Serrano, I., Aper, J., De Swaef, T., Ghesquiere, A., Quataert, P., Roldán-Ruiz, I., Janssens, I. A., Ruysschaert, G., and Lootens, P. (2021). Improving accuracy of herbage yield predictions in perennial ryegrass with uav-based structural and spectral data fusion and machine learning. *Remote Sensing*, 13(17):3459.

[163] Qi, J., Liu, X., Liu, K., Xu, F., Guo, H., Tian, X., Li, M., Bao, Z., and Li, Y. (2022). An improved yolov5 model based on visual attention mechanism: Application to recognition of tomato virus disease. *Computers and electronics in agriculture*, 194:106780.

[164] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788.

[165] Rehman, T. U., Ma, D., Wang, L., Zhang, L., and Jin, J. (2020). Predictive spectral analysis using an end-to-end deep model from hyperspectral images for high-throughput plant phenotyping. *Computers and Electronics in Agriculture*, 177:105713.

[166] Rellán-Álvarez, R., Lobet, G., Lindner, H., Pradier, P.-L., Sebastian, J., Yee, M.-C., Geng, Y., Trontin, C., LaRue, T., Schrager-Lavelle, A., et al. (2015). Glo-roots: an imaging platform enabling multidimensional characterization of soil-grown root systems. *elife*, 4:e07597.

[167] Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

[168] Reynolds, M. and Trethowan, R. (2007). Physiological interventions in breeding for adaptation to abiotic stress. In *Scale and complexity in plant systems research*, pages 129–146. Springer.

[169] Rice, D. B., Kloda, L. A., Levis, B., Qi, B., Kingsland, E., and Thombs, B. D. (2016). Are medline searches sufficient for systematic reviews and meta-analyses of the diagnostic accuracy of depression screening tools? a review of meta-analyses. *Journal of Psychosomatic Research*, 87:7–13.

[170] Rolland, V., Farazi, M. R., Conaty, W. C., Cameron, D., Liu, S., Petersson, L., and Stiller, W. N. (2022). Hairnet: a deep learning model to score leaf hairiness, a key phenotype for cotton fibre yield, value and insect resistance. *Plant Methods*, 18(1):8.

[171] Rong, J., Zhou, H., Zhang, F., Yuan, T., and Wang, P. (2023). Tomato cluster detection and counting using improved yolov5 based on rgb-d fusion. *Computers and Electronics in Agriculture*, 207:107741.

[172] Rouphael, Y., Spíchal, L., Panzarová, K., Casa, R., and Colla, G. (2018). High-throughput plant phenotyping for developing novel biostimulants: from lab to field or from field to lab? *Frontiers in plant science*, 9:408288.

[173] Roy, A. M. and Bhaduri, J. (2022). Real-time growth stage detection model for high degree of occultation using densenet-fused yolov4. *Computers and Electronics in Agriculture*, 193:106694.

[174] Ruiz-Ponce, P., Ortiz-Perez, D., Garcia-Rodriguez, J., and Kiefer, B. (2023). Poseidon: A data augmentation tool for small object detection datasets in maritime environments. *Sensors*, 23(7):3691.

[175] Ruparelia, S., Jethva, M., and Gajjar, R. (2022). Real-time tomato detection, classification, and counting system using deep learning and embedded systems. In *Proceedings of the International e-Conference on Intelligent Systems and Signal Processing: e-ISSP 2020*, pages 511–522. Springer.

[176] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252.

[177] Santos, T. T., De Souza, L. L., dos Santos, A. A., and Avila, S. (2020). Grape detection, segmentation, and tracking using deep neural networks and three-dimensional association. *Computers and Electronics in Agriculture*, 170:105247.

[178] Scherer, D., Müller, A., and Behnke, S. (2010). Evaluation of pooling operations in convolutional architectures for object recognition. In Diamantaras, K., Duch, W., and Iliadis, L., editors, *Artificial Neural Networks–ICANN 2010*, pages 92–101. Springer Berlin Heidelberg, Berlin, Heidelberg.

[179] Serbin, S. P., Singh, A., Desai, A. R., Dubois, S. G., Jablonski, A. D., Kingdon, C. C., Kruger, E. L., and Townsend, P. A. (2015). Remotely estimating photosynthetic capacity, and its response to temperature, in vegetation canopies using imaging spectroscopy. *Remote Sensing of Environment*, 167:78–87.

[180] Shafiekhani, A., Kadam, S., Fritschi, F. B., and DeSouza, G. N. (2017). Vinobot and vinoculer: Two robotic platforms for high-throughput field phenotyping. *Sensors*, 17(1):214.

[181] Shen, C., Liu, L., Zhu, L., Kang, J., Wang, N., and Shao, L. (2020). High-throughput in situ root image segmentation based on the improved deeplabv3+ method. *Frontiers in Plant Science*, 11:576791.

[182] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[183] Singh, A. et al. (2018a). Review of machine learning techniques for plant stress detection. *Trends in Plant Science*.

[184] Singh, A., Ganapathysubramanian, B., Singh, S., and Sarkar, S. (2016). High-throughput phenotyping: a new frontier in agriculture. *Trends in Plant Science*, 21(10):861–873.

[185] Singh, A., Ganapathysubramanian, B., Singh, S., and Sarkar, S. (2018b). Machine learning for plant phenotyping: a review. *Plant Science*, 273:61–71.

[186] Solimani, F., Cardellicchio, A., Dimauro, G., Mininni, A., Calabritto, M., Di Biase, R., Petrozza, A., Summerer, S., Cellini, F., and Renò, V. (2024a). Enhancing small object detection in the yolov8 model: A comprehensive analysis of the optimized model head adaptations. *IEEE Journal*.

[187] Solimani, F., Cardellicchio, A., Dimauro, G., Petrozza, A., Summerer, S., Cellini, F., and Renò, V. (2024b). Optimizing tomato plant phenotyping detection: Boosting yolov8 architecture to tackle data complexity. *Computers and Electronics in Agriculture*, 218:108728.

[188] Solimani, F., Cardellicchio, A., Nitti, M., Lako, A., Dimauro, G., and Renò, V. (2023). A systematic review of effective hardware and software factors affecting high-throughput plant phenotyping. *Information*, 14(4):214.

[189] Steduto, P., Faurès, J.-M., Hoogeveen, J., Winpenny, J., and Burke, J. (2012). Coping with water scarcity: an action framework for agriculture and food security. *Rome: Food and Agriculture Organization of the United Nations*.

[190] Suo, R., Gao, F., Zhou, Z., Fu, L., Song, Z., Dhupia, J., Li, R., and Cui, Y. (2021). Improved multi-classes kiwifruit detection in orchard to avoid collisions during robotic picking. *Computers and Electronics in Agriculture*, 182:106052.

[191] Tan, M., Pang, R., and Le, Q. V. (2020). Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10781–10790.

[192] Tardieu, F., Cabrera-Bosquet, L., Pridmore, T., and Bennett, M. (2017). Plant phenomics, from sensors to knowledge. *Current Biology*, 27(15):R770–R783.

[193] Teramoto, S. and Uga, Y. (2020). A deep learning-based phenotypic analysis of rice root distribution from field images. *Plant phenomics*.

[194] Tracy, S. R., Nagel, K. A., Postma, J. A., Fassbender, H., Wasson, A., and Watt, M. (2020). Crop improvement from phenotyping roots: highlights reveal expanding opportunities. *Trends in plant science*, 25(1):105–118.

[195] Tyagi, A. C. (2016). Towards a second green revolution. *Irrigation and Drainage*, 65(4):388–389.

[196] Ubbens, J. R. and Stavness, I. (2017). Deep plant phenomics: a deep learning platform for complex plant phenotyping tasks. *Frontiers in plant science*, 8:1190.

[197] Underwood, J., Wendel, A., Schofield, B., McMurray, L., and Kimber, R. (2017). Efficient in-field plant phenomics for row-crops with an autonomous ground vehicle. *Journal of field robotics*, 34(6):1061–1083.

[198] Van De Looverbosch, T., Vandenbussche, B., Verboven, P., and Nicolaï, B. (2022). Nondestructive high-throughput sugar beet fruit analysis using x-ray ct and deep learning. *Computers and Electronics in Agriculture*, 200:107228.

[199] Varshney, R. K., Graner, A., and Sorrells, M. E. (2005). Genic microsatellite markers in plants: features and applications. *TRENDS in Biotechnology*, 23(1):48–55.

[200] Vasconez, J. P., Delpiano, J., Vougioukas, S., and Cheein, F. A. (2020). Comparison of convolutional neural networks in fruit detection and counting: A comprehensive evaluation. *Computers and Electronics in Agriculture*, 173:105348.

[201] Walsh, J. J., Mangina, E., and Negrão, S. (2024). Advancements in imaging sensors and ai for plant stress detection: A systematic literature review. *Plant Phenomics*, 6:0153.

[202] Walter, A., Liebisch, F., and Hund, A. (2015). Plant phenotyping: from bean weighing to image analysis. *Plant methods*, 11:1–11.

[203] Wang, C.-Y., Bochkovskiy, A., and Liao, H.-Y. M. (2023a). Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7464–7475.

[204] Wang, C.-Y., Liao, H.-Y. M., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W., and Yeh, I.-H. (2020). Cspnet: A new backbone that can enhance learning capability of cnn. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 390–391.

[205] Wang, M., Fu, B., Fan, J., Wang, Y., Zhang, L., and Xia, C. (2023b). Sweet potato leaf detection in a natural scene based on faster r-cnn with a visual attention mechanism and diou-nms. *Ecological Informatics*, 73:101931.

[206] Wang, P., Niu, T., and He, D. (2021). Tomato young fruits detection method under near color background based on improved faster r-cnn with attention mechanism. *Agriculture*, 11(11):1059.

[207] Wang, T., Rostamza, M., Song, Z., Wang, L., McNickle, G., Iyer-Pascuzzi, A. S., Qiu, Z., and Jin, J. (2019). Segroot: A high throughput segmentation method for root image analysis. *Computers and electronics in agriculture*, 162:845–854.

[208] Wang, X., Wu, Z., Jia, M., Xu, T., Pan, C., Qi, X., and Zhao, M. (2023c). Lightweight sm-yolov5 tomato fruit detection algorithm for plant factory. *Sensors*, 23(6):3336.

[209] Wang, Y., Yan, G., Meng, Q., Yao, T., Han, J., and Zhang, B. (2022). Dse-yolo: Detail semantics enhancement yolo for multi-stage strawberry detection. *Computers and electronics in agriculture*, 198:107057.

[210] Wang, Y.-H. and Su, W.-H. (2022). Convolutional neural networks in computer vision for grain crop phenotyping: A review. *Agronomy*, 12(11):2659.

[211] White, J. W., Andrade-Sanchez, P., Gore, M. A., Bronson, K. F., Coffelt, T. A., Conley, M. M., Feldmann, K. A., French, A. N., Heun, J. T., Hunsaker, D. J., et al. (2012). Field-based phenomics for plant genetics research. *Field Crops Research*, 133:101–112.

[212] Wilf, P., Zhang, S., Chikkerur, S., Little, S. A., Wing, S. L., and Serre, T. (2016). Computer vision cracks the leaf code. *Proceedings of the National Academy of Sciences*, 113(12):3305–3310.

[213] Wilson, A. C., Roelofs, R., Stern, M., Srebro, N., and Recht, B. (2017). The marginal value of adaptive gradient methods in machine learning. *Advances in neural information processing systems*, 30.

[214] Wu, C., Zeng, R., Pan, J., Wang, C. C., and Liu, Y.-J. (2019). Plant phenotyping by deep-learning-based planner for multi-robots. *IEEE Robotics and Automation Letters*, 4(4):3113–3120.

[215] Wu, Y. and Liu, L. (2023). Selecting and composing learning rate policies for deep neural networks. *ACM Transactions on Intelligent Systems and Technology*, 14(2):1–25.

[216] Xie, C. and Yang, C. (2020). A review on plant high-throughput phenotyping traits using uav-based sensors. *Computers and Electronics in Agriculture*, 178:105731.

[217] Xu, R. and Li, C. (2022). A review of high-throughput field phenotyping systems: focusing on ground robots. *Plant Phenomics*.

[218] Xu, Z., York, L. M., Seethepalli, A., Bucciarelli, B., Cheng, H., and Samac, D. A. (2022). Objective phenotyping of root system architecture using image augmentation and machine learning in alfalfa (medicago sativa l.). *Plant Phenomics*.

[219] Xuan, K., Deng, L., Xiao, Y., Wang, P., and Li, J. (2023). So-yolov5: Small object recognition algorithm for sea cucumber in complex seabed environment. *Fisheries Research*, 264:106710.

[220] Yamamoto, K., Guo, W., and Ninomiya, S. (2016). Node detection and internode length estimation of tomato seedlings based on image analysis and machine learning. *Sensors*, 16(7):1044.

[221] Yamamoto, K., Guo, W., Yoshioka, Y., and Ninomiya, S. (2014). On plant detection of intact tomato fruits using image analysis and machine learning methods. *Sensors*, 14(7):12191–12206.

[222] Yang, G., Wang, J., Nie, Z., Yang, H., and Yu, S. (2023). A lightweight yolov8 tomato detection algorithm combining feature enhancement and attention. *Agronomy*, 13(7):1824.

[223] Yang, S., Wang, W., Zhang, Z., Li, Y., and Zhang, C. (2017). Unmanned aerial vehicles (uavs) for remote sensing applications: A review. *International Journal of Applied Earth Observation and Geoinformation*, 62:139–153.

[224] Yang, W., Duan, L., Chen, G., Xiong, L., and Liu, Q. (2013). Plant phenomics and high-throughput phenotyping: accelerating rice functional genomics using multidisciplinary technologies. *Current opinion in plant biology*, 16(2):180–187.

[225] Yu, S., Fan, J., Lu, X., Wen, W., Shao, S., Guo, X., and Zhao, C. (2022). Hyperspectral technique combined with deep learning algorithm for prediction of phenotyping traits in lettuce. *Frontiers in plant science*, 13:927832.

[226] Yuan, W. and Gao, K.-X. (2020). Eadam optimizer: How ε impact adam. *arXiv preprint arXiv:2011.02150*.

[227] Zarco-Tejada, P. J., Camino, C., and Beck, P. S. (2012a). Hyperspectral indices and model simulation for chlorophyll estimation in open-canopy tree crops. *Remote Sensing of Environment*, 121:375–388.

[228] Zarco-Tejada, P. J., González-Dugo, V., and Berni, J. A. (2012b). Fluorescence, temperature and narrow-band indices acquired from a uav platform for water stress detection using a micro-hyperspectral imager and a thermal camera. *Remote sensing of environment*, 117:322–337.

[229] Zeng, G., Birchfield, S. T., and Wells, C. E. (2008). Automatic discrimination of fine roots in minirhizotron images. *New Phytologist*, 177(2):549–557.

[230] Zeng, T., Li, S., Song, Q., Zhong, F., and Wei, X. (2023). Lightweight tomato real-time detection method based on improved yolo and mobile deployment. *Computers and electronics in agriculture*, 205:107625.

[231] Zenkl, R., Timofte, R., Kirchgessner, N., Roth, L., Hund, A., Van Gool, L., Walter, A., and Aasen, H. (2022). Outdoor plant segmentation with deep learning for high-throughput field phenotyping on a diverse wheat dataset. *Frontiers in plant science*, 12:774068.

[232] Zhang, C., Marzougui, A., and Sankaran, S. (2020). High-resolution satellite imagery applications in crop phenotyping: An overview. *Computers and Electronics in Agriculture*, 175:105584.

[233] Zhang, J., Zhang, J., Zhou, K., Zhang, Y., Chen, H., and Yan, X. (2023). An improved yolov5-based underwater object-detection framework. *Sensors*, 23(7):3693.

[234] Zhang, Y. et al. (2021). Temporal deep learning models for plant stress monitoring. *Plant Phenomics*.

[235] Zhao, H., Wang, N., Sun, H., Zhu, L., Zhang, K., Zhang, Y., Zhu, J., Li, A., Bai, Z., Liu, X., et al. (2022). Rhizopot platform: A high-throughput in situ root phenotyping platform with integrated hardware and software. *Frontiers in plant science*, 13:1004904.

[236] Zhao, Y., Zheng, B., Chapman, S. C., Laws, K., George-Jaeggli, B., Hammer, G. L., Jordan, D. R., and Potgieter, A. B. (2021). Detecting sorghum plant and head features from multispectral uav imagery. *Plant Phenomics*.

[237] Zheng, T., Jiang, M., Li, Y., and Feng, M. (2022). Research on tomato detection in natural environment based on rc-yolov4. *Computers and Electronics in Agriculture*, 198:107029.

[238] Zhou, S., Chai, X., Yang, Z., Wang, H., Yang, C., and Sun, T. (2021). Maize-ias: a maize image analysis software using deep learning for high-throughput plant phenotyping. *Plant methods*, 17(1):48.

[239] Zhu, C., Hu, Y., Mao, H., Li, S., Li, F., Zhao, C., Luo, L., Liu, W., and Yuan, X. (2021). A deep learning-based method for automatic assessment of stomatal index in wheat microscopic images of leaf epidermis. *Frontiers in Plant Science*, 12:716784.

[240] Zhu, J.-K. (2016). Abiotic stress signaling and responses in plants. *Cell*, 167(2):313–324.