

Master's Report

In order to obtain a Master's degree in Computer Science

Major: Computer Systems and Software (SIL)

Drone and Satellite-Based Remote Sensing for Early Wheat Disease Detection: A Literature Review.

Réalisé par :

ABBACI Zoulikha
kz_abbaci@esi.dz

BELLALI Amira
ka_bellali@esi.dz

Encadré par:

Dr. BESSAH Naima
n_bessah@esi.dz

Dr. SEHAD Abdenour
a_sehad@esi.dz

Acknowledgments

We would like to express our sincere gratitude to everyone who supported us throughout the completion of this work. Our thanks go especially to our supervisors, professors, and colleagues whose guidance and encouragement were invaluable. We are also grateful to our families and friends for their constant support and motivation.

Abstract

here you can make your abstract

Keywords — Smart Agriculture, Deep Learning, Computer Vision, Wheat Diseases, Pest Detection.

Résumé

here make your summery in french

Mots clés — Agriculture intelligente, apprentissage profond, vision par ordinateur, maladies du blé, détection des parasites.

ملخص

ملخصك باللغة العربية

الكلمات المفتاحية – الزراعة الذكية، التعلم العميق، الرؤية الحاسوبية، أمراض القمح، كشف الآفات الحشرية.

Contents

Cover page	
Abstract	ii
Résumé	i
مُلخص	iii
Contents	vii
List of figures	x
List of Table	xi
General introduction	1
1 Wheat Diseases and Insect Pests – Challenges, Impacts, and Smart Agriculture Solutions	2
1.1 Introduction	2
1.2 The Motivation to Protect Wheat	2
1.3 Wheat Diseases: Types and Impacts	3
1.3.1 Leaf Rust (Brown Rust)	4
1.3.2 Stem Rust (Black Rust)	5
1.3.3 Stripe Rust (Yellow Rust)	5

1.3.4	Blotch Diseases	6
1.3.5	Fusarium Head Blight (FHB)	6
1.3.6	Loose Smut	7
1.3.7	Powdery mildew	7
1.3.8	Common Root Rot	8
1.4	Common Insect Pests in Wheat Cultivation	8
1.4.1	Aphids	9
1.4.2	Cereal leaf beetle	9
1.4.3	Armyworm	9
1.4.4	Pod Borer	10
1.4.5	Brown wheat mite	10
1.4.6	Pink stem borer	11
1.4.7	Sawfly	11
1.4.8	Slugs, Snails, Grasshoppers, and Crickets	12
1.4.9	Wireworm	12
1.5	Enhancing Wheat Disease Control Strategies	13
1.5.1	Usual Instruments (Classic Methods)	13
1.5.2	Technological Tools	14
1.6	Challenges Facing the Integration of Smart Agricultural Systems . .	14
1.7	Conclusion	15
2	Deep Learning for Smart Agriculture	16
2.1	Introduction	16
2.2	Overview of Smart Agriculture	16
2.3	Deep Learning Fundamentals	17
2.3.1	Deep Neural Network Basics (DNN)	17

2.3.2	Learning Process in Deep Neural Networks	19
2.3.3	Model Training Concepts	20
2.3.4	Optimization Methods and Strategies	21
2.3.5	Model Evaluation and Validation	22
2.4	Deep Learning Approaches	22
2.5	Convolutional Neural Networks (CNNs) for Plant Disease Classification	23
2.5.1	Fundamentals of CNNs	23
2.5.2	CNN Architectures for Image Classification	26
2.6	Transfer Learning and Pretrained Models	31
2.6.1	Concept of Transfer Learning	31
2.6.2	Fine-Tuning Strategies for Agricultural Data	33
2.7	Object Detection in Smart Agriculture	33
2.7.1	Key Concepts in Object Detection	34
2.7.2	Key Architectures	36
2.8	Challenges of Deep Learning in Agricultural Contexts	39
2.9	Conclusion	39
3	Integrating Remote Sensing and AI-Based Methods for Detecting Wheat Diseases	41
3.1	Introduction	41
3.2	Imaging Technologies in Remote Sensing	41
3.2.1	RGB Cameras (Red, Green, Blue)	42
3.2.2	Near-Infrared (NIR) Cameras	42
3.2.3	Thermal Infrared Cameras	43
3.2.4	Multispectral Cameras	43
3.2.5	LiDAR Cameras (Light Detection and Ranging)	44

3.2.6	Hyperspectral Cameras	44
3.3	Remote Sensing Platforms	44
3.3.1	Satellite-Based Imaging	44
3.3.2	Aircraft-Based Imaging	45
3.3.3	UAV-Based Imaging	46
3.4	Vegetation Indices	47
3.5	Image processing	47
3.6	Applications of Remote Sensing Data	48
3.7	Workflow for Integrating Remote Sensing and ML/DL in Wheat Disease Detection	49
3.7.1	Features Extraction	50
3.7.2	Statistics-Based Methods	51
3.7.3	Conventional Machine Learning (ML)-Based Methods	52
3.7.4	Deep Learning (DL)-Based Methods	53
3.8	Fusion of Satellite and UAV Data	55
3.8.1	Systematic Categorization of UAV/Satellite Monitoring Methods	56
3.8.2	Challenges of UAV/Satellite Data Fusion	59
3.9	Challenges in Integrating Remote Sensing with ML/DL	59
3.10	Future Perspectives	60
3.11	Conclusion	60
General conclusion		61
Bibliography		62
A Dependencies and libraries		66

List of Figures

1.1	Division of sowing area in the world in 2009 (in million hectares and percentage) Source: FAO, 2011	3
1.2	Taxonomy of wheat diseases [21]	4
1.3	Leaf rust with Brown spores (1), Leaf rust with Black spores(2) [14]	5
1.4	Stem rust [14]	5
1.5	Stripe rust [14]	6
1.6	Symptoms of foliar blotch diseases. (A) Septoria tritici blotch. (B) Tan spot. (C) Septoria nodorum blotch [17]	6
1.7	Symptoms of Fusarium head blight/scab. (A) Early infection signs manifested as a partially bleached wheat head. (B) Advanced infection of Fusarium graminearum [17]	7
1.8	Loose smut [14]	7
1.9	Powdery mildew from Kaggle dataset ‘Wheat Plant Diseases.’	8
1.10	Common root rot from Kaggle dataset ‘Wheat Plant Diseases.’	8
1.11	Aphids on wheat grains (left) and on leaves (right) [16]	9
1.12	Adult (left) and larvae (right) of cereal leaf beetle on wheat crop [16]	9
1.13	Armyworm in wheat field [16]	10
1.14	Pod borer larva (left) and adult (right) from the wheat field [16]	10
1.15	Brown wheat mite on wheat leaves [27]	11
1.16	Pink stem borer [16]	11
1.17	Sawfly from Kaggle dataset “Pest Dataset.”	12

1.18	Grasshoppers from kaggle dataset “Pest Dataset.”	12
1.19	Wireworm in wheat field [16]	13
2.1	Biological Neuron Structure and Its Mathematical Model Representation [32].	17
2.2	The structure of a Neural Network in the binary classification task.	18
2.3	The structure of the DNN [47].	20
2.4	Architecture of CNN [45].	24
2.5	The primary calculations executed at each step of the convolutional layer [4].	25
2.6	Three types of pooling operations [4].	25
2.7	The structure of the CNN [9].	26
2.8	The architecture of VGG [4].	27
2.9	The basic structure of Google Block [4].	27
2.10	The basic block diagram for the Xception block architecture [4].	28
2.11	Residual module diagram [15].	29
2.12	The architecture of DenseNet Network [4].	30
2.13	Architecture of EfficientNet-B0 with MBConv as Basic building blocks [2].	30
2.14	Learning process of transfer learning [53].	32
2.15	Integration of the custom CNN with transfer learning networks [26].	32
2.16	Main idea of YOLO [60].	36
2.17	Flowchart of R-CNN [60].	37
2.18	An illustration of the Faster R-CNN model [46].	38
2.19	Architecture of SSD [33].	39
3.1	Image acquisition techniques [19].	42

3.2	RGB original image (a) and RGB image after removal of soil background (b) [35].	42
3.3	The UAV and its sensors [35].	43
3.4	Examples of acquired thermal images from FLIR Tau 2 and WIRIS 2nd Gen at two different UAV flight heights. RGB images acquired by the RGB cameras mounted on the same UAVs were also provided. “T” in the graphs refers to the measured temperature by the corresponding thermal cameras [54].	43
3.5	Unique optical reflectance signature differences of blue, green, red, and near-infrared light emitted from dead, stressed, and healthy plant tissue [42].	47
3.6	Standard Workflow for Generating UAV-Derived Imagery and Data Products [42].	48
3.7	Classification of crop disease detection approaches using UAV-based remote sensing. The elements within the dotted box represent the image features utilized across one or more of the listed approaches [48].	50
3.8	General Workflow of Conventional ML-Based Crop Disease Detection Using UAV Imagery [48].	52
3.9	General Workflow of DL-based crop disease detection using UAV imagery [48].	54
3.10	The distribution of existing works based on (a) sensors and (b) flight altitudes used in UAV image acquisition. Note that M-RGB denotes modified RGB sensors [48].	55
3.11	Hierarchical decision tree for categorizing UAV/Satellite strategies [3].	56
3.12	Diagram of data comparison strategy [3].	57
3.13	Diagram of the Multiscale explanation strategy [3].	57
3.14	Diagram of Model calibration strategy [3].	58
3.15	Diagram of Data fusion strategy [3].	58

Contents

General introduction

The introduction goes here.

Chapter 1

Wheat Diseases and Insect Pests – Challenges, Impacts, and Smart Agriculture Solutions

1.1 Introduction

Wheat is a crucial global crop, but its production is threatened by various diseases and insect pests, leading to significant yield losses. Traditional detection and control methods are often ineffective, highlighting the need for improved solutions.

This chapter explores common wheat diseases and pests, their impact on agriculture, and the importance of timely detection. It also discusses strategies for enhancing disease control and the challenges of implementing smart agricultural technologies.

1.2 The Motivation to Protect Wheat

Wheat is an ancient and vital food crop that provides energy and feeds billions of people around the world (see Figure 1.1). Its demand is growing quickly because it's used in many affordable food products and plays a big role in global food security. The FAO (Food and Agriculture Organization) estimates that by 2050, the world will need about 840 million tonnes of wheat, up from 642 million tonnes today [49]. This doesn't even include the extra needs like animal feed or the impact of climate change.

To meet this growing demand, developing countries need to increase wheat production by 77%, mostly by improving how much wheat is grown on the same land [49]. But this is becoming harder, as wheat productivity is slowing down and diseases are becoming a bigger problem. If we don't manage pests and diseases properly, wheat production could fall short of what the world needs.

That's why it's important to invest in research, use better farming methods, and grow

disease-resistant wheat. Protecting this essential crop is key to making sure we have enough food for the future.

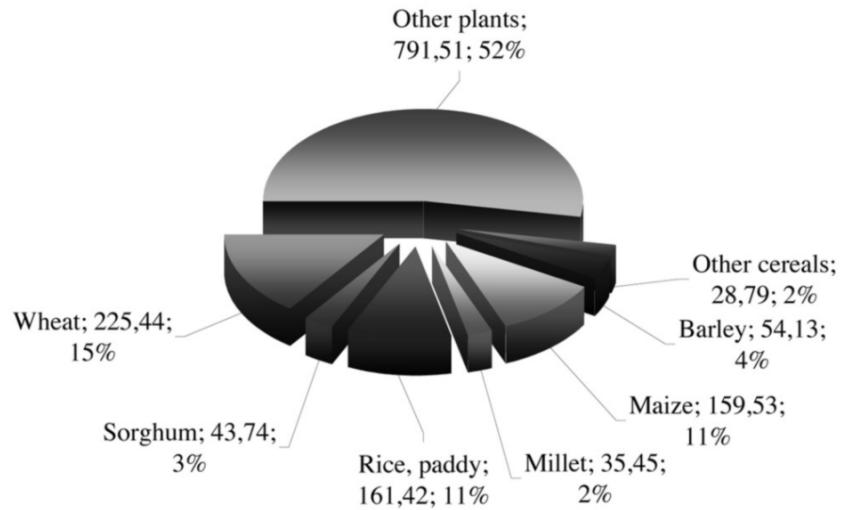


Figure 1.1: Division of sowing area in the world in 2009 (in million hectares and percentage) Source: FAO, 2011

1.3 Wheat Diseases: Types and Impacts

Wheat diseases caused by fungi are influenced by several factors, including plant resistance, spore density, temperature, and environmental conditions, especially the presence of moisture on plant surfaces, which facilitates infection. While some fungi are host-specific, others can infect a wide range of plants. Symptoms can differ greatly, making accurate identification essential. Researchers primarily rely on fungal morphology for diagnosis. A clear understanding of these diseases is key to effective management and control. The following classification (Figure 1.2) outlines the major wheat diseases, grouped by their causes and the plant parts they affect.

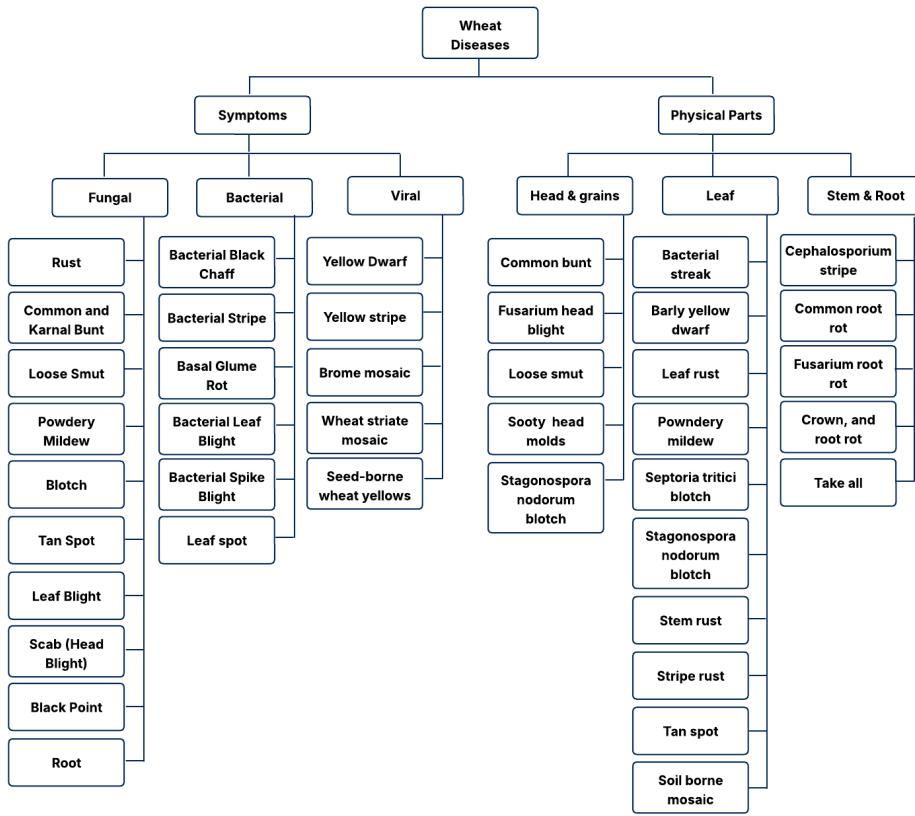


Figure 1.2: Taxonomy of wheat diseases [21]

1.3.1 Leaf Rust (Brown Rust)

duveiller2012wheat

Leaf rust, caused by *Puccinia triticina*, appears as small, circular, orange to brown pustules on the upper surfaces of leaves and leaf sheaths. It spreads through wind-borne spores and develops quickly in moist conditions at around 20°C. New spores form every 10–14 days if conditions are favorable. As plants mature or conditions worsen, black spores may appear (as shown in Figure 1.3). This disease affects wheat, triticale, and related grasses and is common in temperate cereal-growing regions. Severe infections reduce grain yield, kernel number, weight, and quality [14].

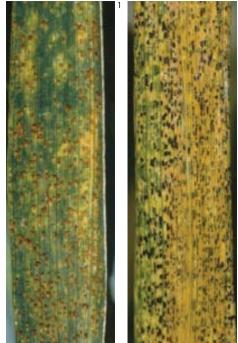


Figure 1.3: Leaf rust with Brown spores (1), Leaf rust with Black spores(2) [14]

1.3.2 Stem Rust (Black Rust)

Stem rust, caused by *Puccinia graminis*, appears as dark reddish-brown pustules on leaves, stems, and spikes (as observed in Figure 1.4). Light infections show scattered pustules, while severe cases cause them to merge. Before pustules form, small flecks may appear, and infected areas feel rough. The disease spreads through wind-borne spores and develops quickly in moist conditions with temperatures around 20°C. New spores can form in 10–15 days. It affects wheat, barley, triticale, and related grasses and is common in temperate cereal regions. Severe infections can reduce grain weight and quality and, in extreme cases, lead to total crop loss [14].



Figure 1.4: Stem rust [14]

1.3.3 Stripe Rust (Yellow Rust)

Stripe rust, caused by *Puccinia striiformis*, appears as yellow to orange-yellow pustules forming narrow stripes on leaves, leaf sheaths, necks, and glumes (as seen in Figure 1.5). It spreads through wind-borne spores and develops quickly in moist conditions at temperatures between 10–20°C but slows down above 25°C. Severe infections reduce grain yield, kernel number, weight, and quality [14].



Figure 1.5: Stripe rust [14]

1.3.4 Blotch Diseases

The blotch diseases, which include *Septoria tritici* blotch (STB), *Septoria nodorum* blotch (SNB), and tan spot (TS) (as presented in Figure 1.6), are caused by the Ascomycete fungi *Zymoseptoria tritici*, *Parastagonospora nodorum*, and *Pyrenophora tritici-repentis*, respectively [17].

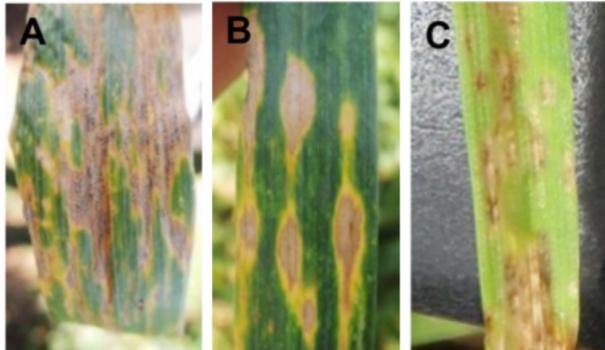


Figure 1.6: Symptoms of foliar blotch diseases. (A) *Septoria tritici* blotch. (B) Tan spot. (C) *Septoria nodorum* blotch [17]

1.3.5 Fusarium Head Blight (FHB)

Fusarium head blight (FHB), also known as wheat scab or ear blight, is a major disease of wheat caused primarily by the Ascomycete fungus *Fusarium graminearum* (Fg). It can also be caused by other regional *Fusarium* species [17]. Fusarium head blight appears as dark, oily florets with pinkish spores (as seen in Figure 1.7). Infected kernels may be covered in white fungal growth. The disease spreads in warm, humid conditions (10–28°C), infecting spikes during flowering and spreading between florets. It affects all small grain cereals and is found in most soils and crop residues. Severe infections can reduce yields by over 50% and lower grain quality. Contaminated grain may contain harmful mycotoxins, making it unsafe for humans and animals [14].



Figure 1.7: Symptoms of Fusarium head blight/scab. (A) Early infection signs manifested as a partially bleached wheat head. (B) Advanced infection of *Fusarium graminearum* [17]

1.3.6 Loose Smut

Loose smut, caused by *Ustilago tritici*, replaces wheat spikes with black fungal spores (as observed in Figure 1.8), which are later dispersed by wind. The fungus infects wheat flowers and stays dormant in kernels until germination. It then grows with the plant, destroying floral parts at flowering. The disease thrives in cool, humid conditions and is found wherever wheat is grown. Yield losses depend on infection levels, usually below 1% but sometimes reaching 30% [14].



Figure 1.8: Loose smut [14]

1.3.7 Powdery mildew

Powdery mildew (as displayed in Figure 1.9) caused by *Blumeria graminis f. sp. tritici* affects wheat globally, particularly in cool, dry climates, and can cause yield losses ranging from 10% to 40%, with severe cases leading to seedling or tiller death [50].



Figure 1.9: Powdery mildew from Kaggle dataset ‘Wheat Plant Diseases.’

1.3.8 Common Root Rot

Common root rot, caused by *Cochliobolus sativus*, *Fusarium* spp., and *Pythium* spp., darkens and weakens wheat roots, crowns, and stems (as illustrated by Figure 1.10), sometimes leading to plant lodging and white spikes before maturity. Early infections can cause seedling death. The disease spreads from infected crop debris, thriving in different soil conditions. *sativus* in warm, dry soils and *Fusarium* and *Pythium* in cool, moist soils. Found in temperate regions, it rarely causes major outbreaks but can lead to localized losses due to reduced plant growth and yield [14].



Figure 1.10: Common root rot from Kaggle dataset ‘Wheat Plant Diseases.’

1.4 Common Insect Pests in Wheat Cultivation

Wheat is affected by several insect pests that can seriously reduce yield and quality. Below are some of the most common pests and their impacts:

1.4.1 Aphids

Aphids are nearly transparent, soft bodied sucking insects (as seen in Figure 1.11). When present in sufficient numbers, aphids can cause the yellowing and premature death of leaves. They exude drops of sugary liquid known as “honeydew” [14]. These pests feed on wheat leaves and grain heads, causing leaf rolling, trapped heads, and poor pollination, particularly during early growth stages. They damage crops by sucking sap from leaves, stems, and kernels, and their honeydew secretion promotes black sooty mold, which hampers photosynthesis and leads to 20-80% yield losses [16].



Figure 1.11: Aphids on wheat grains (left) and on leaves (right) [16]

1.4.2 Cereal leaf beetle

Adult beetles are 4-5 mm long and have a black head, light brown thorax, and a shiny blue-green wing cover with parallel lines of small dots (as observed in Figure 1.12). Larvae are a dull to bright yellow color but soon take on the appearance of a slimy, globular, black mass due to the mound of fecal material they produce and accumulate on their backs. The most prominent symptom of cereal leaf beetle infestations is the distinct longitudinal stripes on leaves; these stripes are produced by the feeding of adult beetles and of larvae [14]. Significant yield losses can occur in winter wheat and fall-sown spring wheat. Yield losses of 14% to more than 25% have occurred with natural infestations [14].



Figure 1.12: Adult (left) and larvae (right) of cereal leaf beetle on wheat crop [16]

1.4.3 Armyworm

The armyworm (*Mythimna separata* Walker) is a pest of wheat. Adult moths are stout and pale brown, while larvae have orange, white, and brown stripes along their sides, a broken stripe on their back, and black spots on the top of their prolegs (as seen in Figure 1.13). Major damage is caused by caterpillars, which move in swarms from field to field, feeding on seedling leaves and ear heads, halting plant growth [16].



Figure 1.13: Armyworm in wheat field [16]

1.4.4 Pod Borer

The pod borer (*Helicoverpa armigera*) is a polyphagous pest that attacks crops such as gram, lablab, safflower, chilies, groundnut, tobacco, cotton, and wheat. The caterpillars start brown and later turn greenish with dark broken lines on their sides. The adult moth is medium-sized with brownish or greyish forewings featuring a dark crossband and spots, with a wingspan of 3.7 cm (as shown in Figure 1.14) [16].



Figure 1.14: Pod borer larva (left) and adult (right) from the wheat field [16]

1.4.5 Brown wheat mite

The brown wheat mite is found in major wheat-growing areas, especially in rainfed conditions. Only females exist and lay two types of eggs in the soil red ones in winter (as seen in Figure 1.15) and white-covered ones in early summer. They damage crops by sucking sap, leading to silvery flecks, yellowing leaves, and reduced grain quality. The mites are active during bright daylight and don't form webs. Infestation begins in December-January and persists until maturity, with winter rains hindering their multiplication [27].



Figure 1.15: Brown wheat mite on wheat leaves [27]

1.4.6 Pink stem borer

The pink stem borer (*Sesamia inferens*), an oriental pest from the Noctuidae family (as illustrated in Figure 1.16), is native to regions such as the Indian subcontinent, China, Pakistan, and Southeast Asia. Initially a rice pest, it has adapted to wheat crops in North-Western India due to changes in tillage practices. Larvae feed inside wheat stems, causing significant damage, including "dead hearts" at the tillering stage and "white heads" at the ripening stage, reducing yields by over 11% in India. Damage symptoms in wheat are similar to those in rice [16].



Figure 1.16: Pink stem borer [16]

1.4.7 Sawfly

Sawflies (as seen in Figure 1.17) produce one generation per year, with larvae overwintering in straw. The legless white larvae bore into wheat stems, weakening plants, causing poor head development, and making them prone to lodging. They primarily target wheat, especially fall-sown varieties, though other cereals can be affected. Infestations are usually patchy and inconsistent. The wheat stem sawfly (*Cephus cinctus*) is a major concern, it

can significantly damage crops. While typically not widespread, sawfly infestations can cause severe localized yield losses [14].



Figure 1.17: Sawfly from Kaggle dataset “Pest Dataset.”

1.4.8 Slugs, Snails, Grasshoppers, and Crickets

Slugs, snails, grasshoppers (as referenced in Figure 1.18), and crickets are widespread pests affecting wheat and other plants. They damage crops by chewing leaves, causing a frayed appearance in mature plants. While their presence is often localized, large infestations can significantly impact plant health and yield worldwide [14].



Figure 1.18: Grasshoppers from kaggle dataset “Pest Dataset.”

1.4.9 Wireworm

Wireworms are yellow to brown larvae with six short legs (as seen in Figure 1.19). They primarily feed on wheat kernels by consuming the endosperm and leaving only the seed coat. These pests attack young seedlings, often causing "damping off" symptoms and damaging crops at an early stage. Their presence can significantly impact wheat growth and yield, making timely identification and control essential for effective disease management [16].



Figure 1.19: Wireworm in wheat field [16]

1.5 Enhancing Wheat Disease Control Strategies

Effective wheat disease management combines traditional farming practices with modern technologies. Together, they offer a balanced approach to reducing disease impact and improving crop health.

1.5.1 Usual Instruments (Classic Methods)

These are long-standing approaches that provide the foundation for managing wheat diseases [41]. The following practices have been widely used to reduce disease pressure and support healthy crop development.

- **Crop Rotation:** Rotating wheat with non-host crops reduces the buildup of soil-borne pathogens and interrupts disease cycles.
- **Tillage:** Tillage affects disease development by influencing residue decomposition and soil pathogen levels; conservation tillage can increase some necrotrophic diseases.
- **Healthy Seeds:** Clean, pathogen-free seeds minimize seed-borne disease transmission and ensure strong early crop establishment.
- **Soil Management:** Managing soil pH, structure, and nutrient balance helps prevent stress-related susceptibility and supports healthier root systems.
- **Fertilizer Use:** Balanced fertilization strengthens plant defense mechanisms, while over- or under-fertilization can predispose plants to infection.
- **Diversification of Cultivars and Sowing Dates:** Altering cultivars and planting schedules helps reduce the uniformity that pathogens exploit and spreads risk across environments.

- **Use of Resistant Cultivars:** Cultivars bred for specific, partial, or generalized resistance can significantly reduce disease severity, especially when tailored to local pathogen races.
- **Alternative Eco-Friendly Practices:** Methods like field sanitation, residue management, and proper spacing contribute to reducing pathogen survival and disease spread.

1.5.2 Technological Tools

Complementing traditional methods, the following tools, rooted in smart agriculture, offer modern solutions to enhance the effectiveness and precision of wheat disease management.

- **Remote Sensing:** Remote sensing using unmanned aerial vehicle-mounted multi-spectral sensors enables high-resolution monitoring of wheat canopy characteristics across different growth stages. By analyzing spectral bands (Green, Red, Red Edge, Near Infrared), the system captures critical indicators of plant health, canopy structure, and stress conditions, supporting precise and timely crop management [55].
- **Disease Forecast Modeling:** Weather-based and biological models are used to predict disease outbreaks and support timely interventions [41].
- **Computer Vision:** Enables automated analysis of crop images for monitoring wheat growth, detecting diseases, and assessing yield. It plays a crucial role in real-time decision-making by processing visual data from the field [19].
- **AI and Machine Learning Algorithms:** Used to interpret complex image data, these algorithms support tasks like disease classification, crop health prediction, and optimizing farm operations by learning from patterns in large agricultural datasets [19].
- **Autonomous Robotic Platforms and Drones:** Facilitate efficient field data collection, spraying, and crop monitoring. These tools reduce manual labor and enable precise, targeted interventions across large wheat fields [19].
- **Precision Agriculture Systems:** Precision agriculture systems integrate technologies like the Global Positioning System (GPS) and the Internet of Things (IOT) to manage field variability. These technologies help optimize the use of inputs (e.g., water, fertilizer) and support sustainable, data-driven wheat farming [19].

1.6 Challenges Facing the Integration of Smart Agricultural Systems

Fully automated smart farming faces both technical and practical challenges. A major obstacle is the generalization of computer vision models across diverse field conditions

like lighting, weather, soil, and crop types, which complicates real-time deployment. Robust decision-making in unpredictable outdoor environments also remains difficult, and integrating the full pipeline from image capture to treatment is still under development [19].

On the technical side, communication protocols often support only short distances, limiting scalability. Many devices rely on batteries, reducing operational time. Additionally, processing the large volumes of data generated introduces computational bottlenecks, alongside concerns about privacy, trust, and security in data handling [25].

1.7 Conclusion

This chapter presented an overview of the major wheat diseases and pests, along with the current challenges faced in their detection and management. These challenges highlight the need for more advanced and efficient solutions. In the next chapter, we examine how machine learning and deep learning technologies are applied to improve wheat disease classification and pest detection, supporting the shift toward smarter and more sustainable agricultural practices.

Chapter 2

Deep Learning for Smart Agriculture

2.1 Introduction

Deep learning and computer vision have become indispensable tools in modern agriculture, enabling automated, high-precision analysis of crop health, disease detection, and pest management. By leveraging convolutional neural networks (CNNs), these technologies can process vast amounts of visual data from drone imagery to ground-based cameras, identifying subtle patterns indicative of wheat diseases. This shift from manual scouting to AI-driven monitoring improves scalability, reduces human error, and supports timely interventions, ultimately enhancing yield and sustainability.

In this chapter, we establish the foundational concepts of deep learning and computer vision as applied to agricultural challenges. We begin by examining the core principles of CNNs and their role in feature extraction from crop imagery. Next, we explore transfer learning techniques, which allow pre-trained models to adapt to agricultural datasets with limited labeled examples. The chapter then discusses object detection methods critical for localizing diseases and pests in field conditions. Finally, we address the practical challenges of implementing these solutions, including data variability, model scalability, and real-world deployment constraints in agricultural settings.

2.2 Overview of Smart Agriculture

Smart agriculture integrates advanced technologies such as the Internet of Things (IoT), artificial intelligence (AI), and big data analytics to enhance farming efficiency and sustainability. By leveraging real-time data from sensors, drones, and satellite imagery, smart agricultural systems enable precise monitoring of soil conditions, crop health, and environmental factors [18]. Machine learning and deep learning techniques, particularly computer vision, play a crucial role in automating disease detection, pest control, and yield prediction. These AI-driven solutions reduce dependency on manual labor, optimize resource usage, and facilitate data-driven decision-making. The adoption of smart agriculture not only improves productivity but also promotes sustainable farming practices,

addressing global food security challenges [18].

2.3 Deep Learning Fundamentals

In this section, we introduce the core principles of deep learning, focusing on deep neural networks (DNNs) and their role in learning complex patterns from data. We explore the structure and components of neural networks, the learning process, optimization techniques, and model evaluation methods. Additionally, we highlight regularization strategies to prevent overfitting.

2.3.1 Deep Neural Network Basics (DNN)

Neural networks form the backbone of deep learning, enabling machines to learn patterns and make predictions from data. Inspired by the structure of the human brain, these networks consist of interconnected layers of artificial neurons that hierarchically process information.

Definition of DNN

Before defining deep neural networks, we first need to understand two essential components:

- **Artificial neuron:** An artificial neuron is the basic building block of artificial neural networks, designed based on the structure and functionality of biological neurons. It receives weighted inputs, processes them through a transfer function, and outputs the result. The artificial neuron model simplifies the biological process where information is received through dendrites, processed in the soma, and transmitted via the axon, as shown in Figure 2.1 [32].
- **Layer:** A layer in a neural network is a set of neurons that perform a specific operation on the data [6].

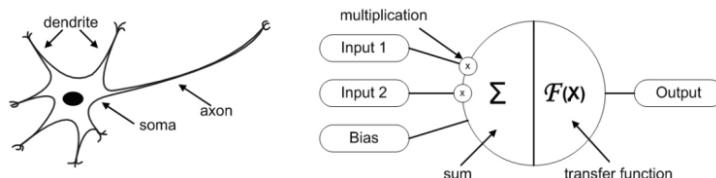


Figure 2.1: Biological Neuron Structure and Its Mathematical Model Representation [32].

By combining multiple layers of interconnected artificial neurons, we arrive at the concept of a Deep Neural Network (DNN). **[A DNN is a neural network]** that contains

multiple hidden layers between the input and output layers. These additional layers enable the network to learn complex patterns and high-level features from data. Each layer transforms its input into a more abstract representation, improving the network's ability to recognize intricate structures and relationships [33].

Structure of DNN

A neural network consists of three main layers [47]:

The input layer: Represents the features of the input data, such as pixel values in an image, denoted as a vector

$$X = [x_1, x_2, \dots, x_n].$$

The hidden layers: Process this input using weighted connections and biases, computed as:

$$z = W \cdot X + b_z \quad (1)$$

$$F(z) = a \quad (2)$$

where:

- W is the weight matrix,
- b is the bias vector,
- z is the pre-activation value,
- $F(z)$ is the activation function applied to z .

Each neuron in the hidden layers applies an activation function to capture complex patterns.

The output layer: Generates the network's prediction, with the number of neurons corresponding to the specific task, such as one neuron for binary classification (as presented in Figure 2.2) or multiple neurons for multiclass classification.

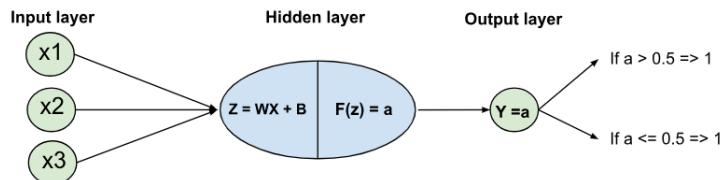


Figure 2.2: The structure of a Neural Network in the binary classification task.

Activation Functions

An activation function (AF) is a mathematical function applied to a neuron's output in a neural network to introduce non-linearity. Without an activation function, a neural

network with multiple layers would behave like a single-layer perceptron, limiting its ability to model complex relationships [13]. Activation functions decide whether a neuron should be activated based on its input.

Table 2.1 summarizes the most commonly used activation functions, their formulas, ranges, and typical use cases in deep neural networks.

Table 2.1: Common Activation Functions in Deep Learning (Dubey et al., 2022)

Activation Function	Formula	Range	Usage
Sigmoid	$\frac{1}{1+e^{-x}}$	[0, 1]	Commonly used in binary classification problems, especially in the output layer of models predicting probabilities.
Tanh (Hyperbolic Tangent)	$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$	[-1, 1]	Often used in hidden layers of neural networks, as it outputs values centered around zero, which helps in reducing bias during optimization.
ReLU (Rectified Linear Unit)	$\max(0, x)$	[0, ∞)	Widely used in hidden layers of deep neural networks due to its simplicity and effectiveness in handling the vanishing gradient problem.
Softmax	$\text{Softmax}(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$	(0, 1) (outputs sum to 1)	Used in the output layer for multi-class classification.

2.3.2 Learning Process in Deep Neural Networks

In the information processing flow within an artificial neuron, several elements, known as parameters, are learned from the training data. These parameters include [28]:

- **Weights:** Weights control the amount of each input feature that passes through the neuron. They represent the coefficients of the connections between neurons in the layers of a neural network. Weights are essential for determining the influence of each input on the output.

- **Biases:** Biases are values added to the outputs of the neurons before applying the activation function. They allow the network to shift the activation function, providing more flexibility to the model.

The learning process in neural networks involves training the model to map input data to desired outputs. This is achieved through the mechanisms of forward propagation and backward propagation, along with optimization techniques that refine the network's parameters (weights and biases) to minimize the error [28]:

- **Forward Propagation:** In forward propagation, the input data passes through the network layer by layer. Each neuron in a layer performs a weighted sum of its inputs, applies an activation function, and passes the result to the next layer. The process continues until the output layer is reached and a prediction is made.
- **Backward Propagation:** Backward propagation (or backpropagation) is used to update the network's weights. After calculating the error (the difference between the predicted and actual output), the error is propagated back through the network. The weights are adjusted based on the gradient of the error with respect to each weight, using optimization algorithms like gradient descent.

Figure 2.3 below provides a visual representation of the learning process in a neural network, illustrating the flow of information from the input layer through multiple hidden layers to the output layer.

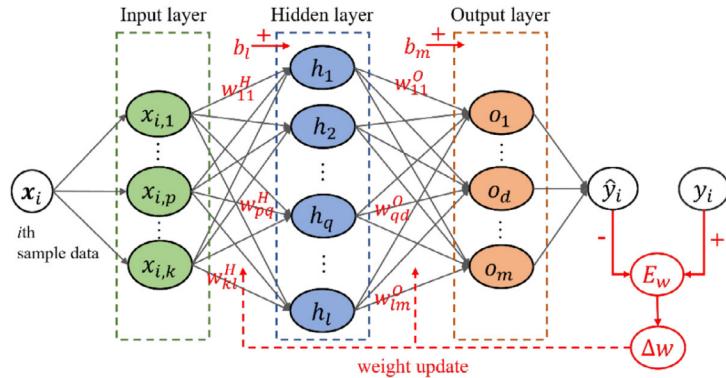


Figure 2.3: The structure of the DNN [47].

2.3.3 Model Training Concepts

Effective model training ensures that neural networks learn patterns in data while minimizing errors. This section highlights essential components of the training process.

Loss Functions Loss functions quantify the error between predicted and actual values, guiding weight updates during optimization [4]. Common loss functions include:

- **Cross-Entropy Loss:** Used for classification tasks to measure the divergence between predicted probabilities and true labels.
- **Mean Squared Error (MSE):** Applied in regression, computing the average squared difference between predicted and actual values.

Overfitting Prevention Overfitting occurs when a model learns noise from training data, reducing generalization to unseen data. Common techniques to mitigate overfitting include:

- **Dropout:** Randomly deactivate neurons during training to enhance robustness.
- **Data Augmentation:** Transform training samples (e.g., rotation, scaling) to increase dataset diversity.

Batch Normalization Batch normalization stabilizes training by normalizing inputs across a mini-batch, reducing internal covariate shift and accelerating convergence.

2.3.4 Optimization Methods and Strategies

Optimization techniques in deep learning are methods used to minimize the loss function during training, improving the model's accuracy. These techniques adjust the model's weights and biases iteratively to find the optimal set of parameters that reduces the error between predicted and actual values [4].

Different optimizers are used to update model weights efficiently:

- **Stochastic Gradient Descent (SGD):** Updates weights based on a small subset (batch) of training data, improving computational efficiency.
- **Adam (Adaptive Moment Estimation):** Combines momentum and adaptive learning rates for faster and more stable convergence.
- **RMSprop:** Uses an adaptive learning rate to prevent oscillations and improve performance on non-stationary objectives.

However, an important consideration in optimization is how to set the learning rate throughout training. While optimizers control how weights are updated, learning rate scheduling adjusts the learning rate over time to further optimize training.

Different scheduling strategies can be used in conjunction with optimizers [59]:

- **Step Decay:** Reduces the learning rate at fixed intervals, allowing for more stable training as the model reaches its optimal solution.

- **Exponential Decay:** Gradually decreases the learning rate across epochs, promoting finer adjustments to the model parameters as the training progresses.
- **Cyclic Learning Rates:** Alternates between a minimum and maximum learning rate, which helps the model escape local minima and enhances exploration of the parameter space.

By combining these scheduling strategies with optimization techniques, the training process can become more efficient and effective, leading to faster and more reliable convergence.

2.3.5 Model Evaluation and Validation

Once a model is trained, it is crucial to assess its performance and ensure that it generalizes well to unseen data. This process is known as **model evaluation and validation**. The primary goal is to measure how well the model performs and to identify any potential overfitting or underfitting issues.

To evaluate a model's performance, several metrics can be used depending on the type of task (classification, regression, etc.) [29]:

- **Accuracy:** For classification tasks, accuracy measures the proportion of correct predictions out of all predictions made. However, accuracy alone can be misleading in imbalanced datasets.
- **Precision and Recall:** In imbalanced classification problems, precision (the proportion of true positive results among all positive predictions) and recall (the proportion of true positive results among all actual positives) are often used in conjunction to provide a clearer view of the model's performance.
- **F1-Score:** The harmonic mean of precision and recall; F1-score balances the two metrics and is especially useful when dealing with imbalanced datasets.
- **Mean Squared Error (MSE):** For regression tasks, MSE calculates the average of the squared differences between predicted and actual values. It penalizes large errors more significantly than smaller ones.
- **R-squared (R²):** For regression, this metric indicates how well the model explains the variability in the data, with a value closer to 1 suggesting a better fit.

2.4 Deep Learning Approaches

Deep learning encompasses a variety of learning paradigms that enable models to extract complex patterns from large volumes of data. These approaches differ based on the nature of the data, the availability of labels, and the interaction mechanisms with the environment. This section presents the four main categories of deep learning methods [4]:

- **Deep Supervised Learning:** This approach involves training a neural network using a labeled dataset, where each input image (e.g., leaf with visible symptoms) is paired with its correct label (e.g., disease type). The model learns to minimize the error between its predictions and the true labels using backpropagation and optimization algorithms. Techniques include Convolutional Neural Networks (CNNs) for spatial feature extraction, Deep Neural Networks (DNNs), and Recurrent Neural Networks (RNNs) like LSTMs and GRUs when dealing with sequential data.
- **Deep Semi-Supervised Learning:** Semi-supervised learning combines a small labeled dataset with a larger pool of unlabeled data, which is common in agricultural settings where annotating plant diseases is costly and time-consuming. Methods like Generative Adversarial Networks (GANs) can generate synthetic labeled images, while RNNs, LSTMs, and Deep Reinforcement Learning (DRL) can help model complex data behavior.
- **Deep Unsupervised Learning:** Unsupervised learning aims to extract meaningful patterns from unlabeled data, such as grouping similar plant images or identifying features without predefined classes. Popular methods include Autoencoders for dimensionality reduction, Restricted Boltzmann Machines, and clustering algorithms. These are useful in exploratory stages or feature extraction before applying a classifier.
- **Deep Reinforcement Learning (DRL):** In DRL, models learn optimal actions through trial and error by interacting with an environment. This could include real-time monitoring systems in agriculture, like automated disease response tools or robotic weeders. Unlike supervised learning, DRL does not require labeled data but instead uses reward signals to adjust its strategy over time.

2.5 Convolutional Neural Networks (CNNs) for Plant Disease Classification

Convolutional Neural Networks (CNNs) have revolutionized image-based plant disease classification by automatically extracting hierarchical features from agricultural images. Unlike traditional machine learning approaches that rely on handcrafted features, CNNs learn spatial patterns directly from raw images, improving classification accuracy. Their ability to recognize disease symptoms from leaf textures and color variations makes them particularly effective in precision agriculture. This section explores the fundamental components of CNNs, their advantages in agricultural applications, and their limitations when applied independently.

2.5.1 Fundamentals of CNNs

Convolutional Neural Networks (CNNs) consist of multiple layers designed to process and learn spatial hierarchies from image data (see Figure 2.4). Their architecture typically includes [4]:

- **Convolutional Layers:** Extract features using small filters (kernels) that detect edges, textures, and patterns. A kernel is a grid of values (weights) initialized randomly at the start of training and adjusted through learning to identify important features.
 - **Input and Kernel Dimensions:** In a CNN layer, each input x is structured in three dimensions: height, width, and depth. The depth corresponds to the number of channels (e.g., an RGB image has three channels). Similarly, the kernels are also three-dimensional, with spatial dimensions (height and width) and a depth matching the input channels. Each kernel has shared parameters—a set of weights and a bias. When applied to the input, these kernels generate a corresponding set of feature maps. These kernels establish local connections by interacting only with small regions of the input at a time, allowing the network to extract patterns such as edges and textures by computing dot products across these regions.
 - **Convolutional Operation:** The convolutional process begins by sliding the kernel across the input image in both horizontal and vertical directions. At each location, the dot product between the kernel and the overlapping region of the input is computed, producing a scalar value that becomes part of the resulting feature map. As this process is repeated across the image, a full feature map is constructed, highlighting areas where the kernel detects specific patterns. Parameters such as stride (controlling how far the kernel moves at each step) and padding (adding borders to the input to preserve edge information) affect the size and coverage of the output feature map. These concepts are illustrated in Figure 2.5.

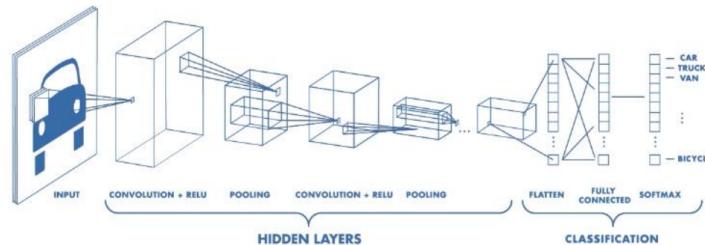


Figure 2.4: Architecture of CNN [45].

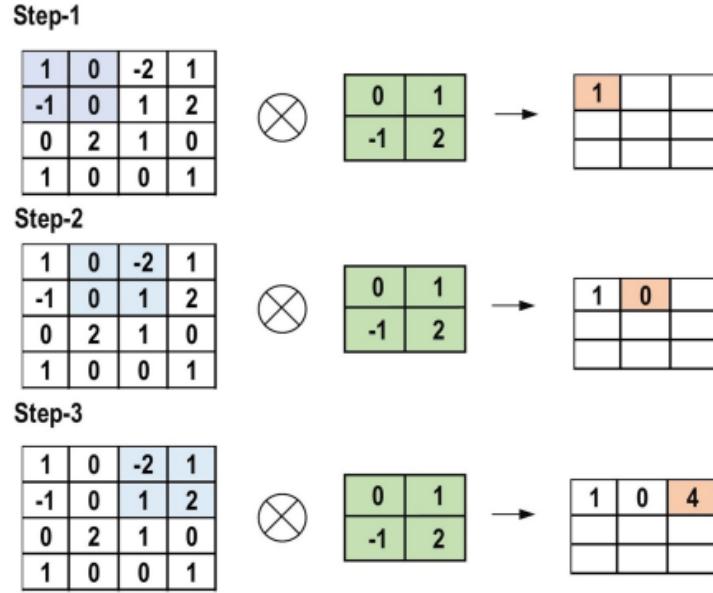


Figure 2.5: The primary calculations executed at each step of the convolutional layer [4].

- **Pooling Layers:** Pooling layers are used to reduce the spatial dimensions of feature maps while preserving the most important information. This reduction helps lower the computational cost and minimizes the risk of overfitting by simplifying the data representation. The pooling operation works by sliding a small filter over the feature map and applying a summary function within each local region (Figure 2.6). Common types of pooling include:
 - **Max Pooling:** Selects the maximum value in each region.
 - **Average Pooling:** Calculates the average value of the region.
 - **Global Average Pooling:** Computes the average across the entire feature map.

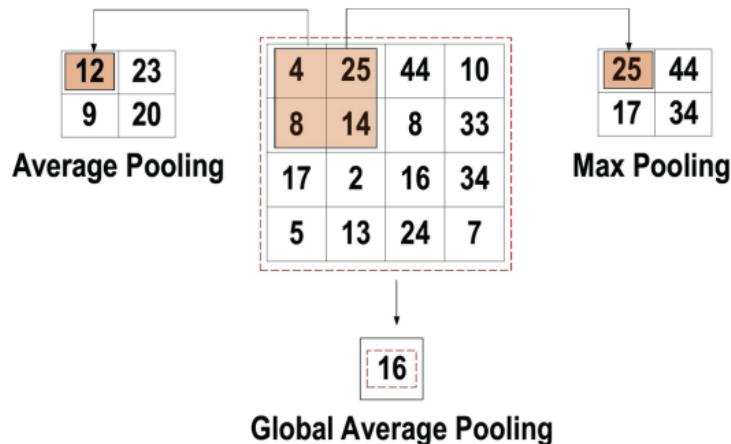


Figure 2.6: Three types of pooling operations [4].

- **Activation Functions:** Introduce non-linearity to help the network learn complex patterns. They must also be differentiable to enable backpropagation during training. CNNs commonly utilize the following activation functions: ReLU (Rectified Linear Unit), Sigmoid, Softmax, and Tanh.
- **Fully Connected Layers:** The Fully Connected (FC) layer is typically found at the end of a CNN architecture and serves as the classifier. In this layer, each neuron is connected to all neurons from the previous layer, following the fully connected approach. It operates similarly to a conventional multi-layer perceptron (MLP) network, which is a type of feed-forward artificial neural network (ANN). The input to the FC layer is a vector created from the feature maps after flattening, which comes from the last pooling or convolutional layer. The output of the FC layer represents the final result of the classification task.

Figure 2.7 below illustrates the general structure of a Convolutional Neural Network (CNN), highlighting its layers, which work together to extract and classify features from input images.

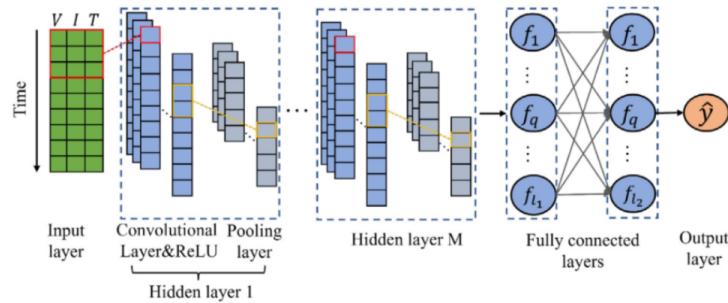


Figure 2.7: The structure of the CNN [9].

2.5.2 CNN Architectures for Image Classification

Over the past decade, numerous Convolutional Neural Network (CNN) architectures have been developed, each introducing unique design principles to improve accuracy, efficiency, and scalability. In the context of image classification, especially for tasks such as plant disease detection, the choice of architecture can significantly influence performance depending on the dataset size, complexity, and computational constraints. This section presents an overview of some of the most influential and widely used CNN architectures:

Visual geometry group network (VGGNet)

Proposed by Simonyan and Zisserman, VGGNet is a convolutional neural network (CNN) architecture widely recognized for its simplicity and strong performance in image recognition tasks.

VGG is characterized by its deep architecture, typically comprising 16 to 19 layers, which significantly enhances its representational power compared to earlier models like

ZFNet and AlexNet. One of its key innovations is the replacement of large convolutional filters (such as 11×11 or 5×5) with multiple stacked 3×3 filters. This strategy maintains an equivalent receptive field while reducing the number of parameters and improving computational efficiency.

In addition, VGG uses 1×1 convolutions to control the model's complexity and includes max pooling layers to progressively reduce the spatial dimensions of the feature maps, as illustrated in Figure 2.8.

Despite its effectiveness, a major drawback of VGGNet is its high computational cost, with around 140 million parameters [4]. Nonetheless, its reliable feature extraction capabilities have made it a popular choice in applications like plant disease classification, especially for detecting early-stage or visually subtle symptoms.

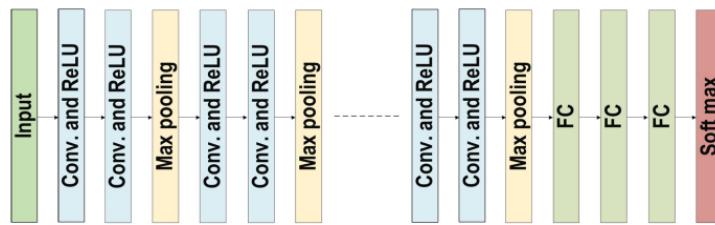


Figure 2.8: The architecture of VGG [4].

Inception Net (GoogLeNet)

Inception Net, introduced by Szegedy et al., uses Inception modules that apply multiple convolutional filters (1×1 , 3×3 , 5×5) in parallel, followed by concatenation (see Figure 2.9). This design captures multi-scale information efficiently while reducing computational cost. The architecture has been successfully applied in plant phenotyping and classification of complex disease patterns.

It replaced standard convolutional layers with micro-neural networks and regulated computation through 1×1 convolutions as bottleneck layers. Sparse connections addressed redundant information by selectively connecting input and output channels, while the global average pooling (GAP) layer reduced parameters from 40 million to just 5 million, enhancing efficiency. Additional features included the RmsProp optimizer, batch normalization, and auxiliary learners to accelerate convergence [4].

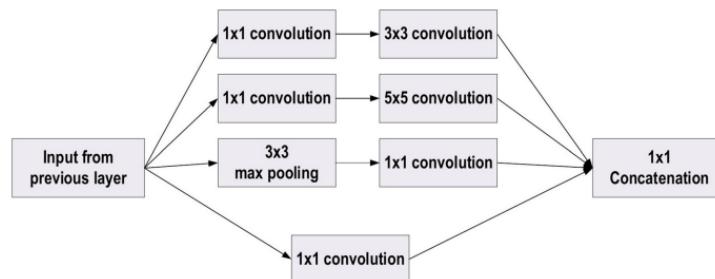


Figure 2.9: The basic structure of Google Block [4].

Xception Net

The Xception model is an extension of the Inception architecture that replaces standard convolutions with depth-wise separable convolutions, significantly improving efficiency. It has been shown to outperform Inception in many tasks, especially with high-resolution agricultural images, by learning spatial and cross-channel correlations separately.

The core concept behind Xception is the modification of the traditional Inception block by making it wider and replacing the standard 3×3 convolution followed by a 1×1 convolution with depthwise separable convolutions, which reduces computational complexity while enhancing performance [4]. An illustration of the basic Xception block structure is presented in Figure 2.10.

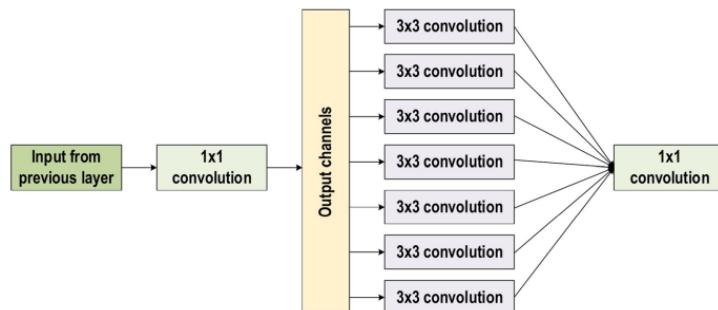


Figure 2.10: The basic block diagram for the Xception block architecture [4].

Residual Networks (ResNet)

ResNet is a deep convolutional neural network architecture developed to facilitate the training of very deep networks, ranging from 18 to 152 layers, by introducing the concept of residual learning through shortcut (or skip) connections that bypass one or more layers.

These residual connections address the degradation problem that often occurs in deeper networks, where increasing depth leads to performance saturation or even degradation. By enabling gradients to flow more efficiently, ResNet makes it easier for the network to learn identity mappings or residual functions, simplifying the overall training process.

The output of a residual block is defined as:

$$H(x) = F(x) + x \quad (3)$$

Where:

- $H(x)$: the output of the residual block,
 - $F(x)$: the residual function to be learned,
 - x : the input passed through the shortcut connection.

This formulation (3) makes it easier to optimize deep networks by learning the difference (residual) between the desired mapping and the identity. Additionally, it helps increase the rank of the weight matrices, enhancing the network's expressiveness and preventing performance degradation.

An illustration of the residual module structure is provided in Figure 2.11.

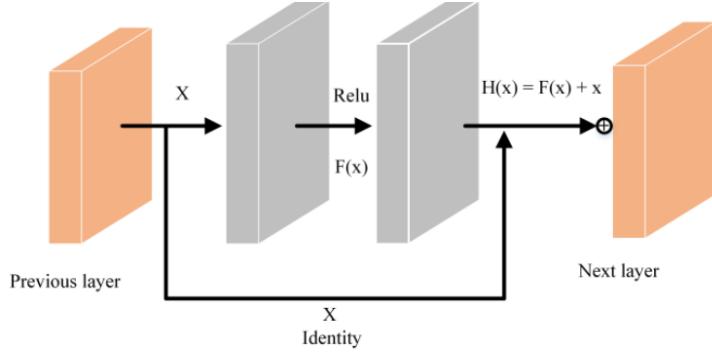


Figure 2.11: Residual module diagram [15].

DenseNet

DenseNet is a convolutional neural network architecture that introduces the concept of dense connectivity, in which each layer is directly connected to every other layer in a feed-forward fashion, as illustrated in Figure 2.12. This unique design enables feature reuse, enhances gradient flow, and significantly reduces the number of parameters compared to traditional CNNs.

Inspired by ResNet and Highway Networks, DenseNet addresses a key limitation in ResNet, where each layer maintains isolated weights and where certain transformations contribute minimal new information. In contrast, DenseNet concatenates the outputs of all preceding layers and feeds them as input to each subsequent layer.

In a DenseNet with l layers, the number of direct connections between layers is given by:

$$\text{Number of connections} = \frac{l(l+1)}{2} \quad (4)$$

This connectivity pattern facilitates richer feature propagation, introduces a regularization effect, and helps mitigate the vanishing gradient problem, thus improving training efficiency and model generalization [4].

Despite the computational cost introduced by the accumulation of feature maps, DenseNet has shown exceptional performance in fine-grained classification tasks, such as distinguishing between subtly different plant disease symptoms, particularly in scenarios with limited training data.

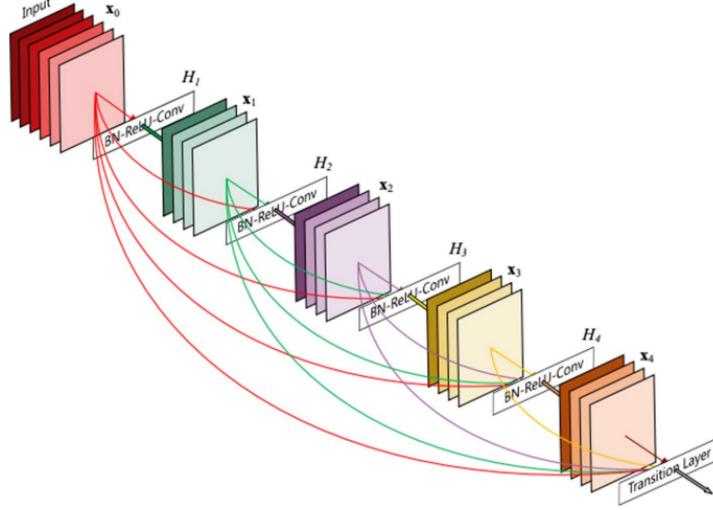


Figure 2.12: The architecture of DenseNet Network [4].

EfficientNet

EfficientNet is a family of convolutional neural networks developed by Google AI that introduces a compound scaling method to efficiently scale deep learning models. Traditional approaches often scale models arbitrarily in one of three dimensions: depth (number of layers), width (number of channels), or input resolution. However, EfficientNet proposes a more balanced and systematic strategy, where all three dimensions are scaled simultaneously and proportionally using a fixed set of scaling coefficients. This compound approach maintains model efficiency while significantly boosting accuracy.

The baseline model, EfficientNet-B0 (see Figure 2.13), is built using Neural Architecture Search (NAS) to optimize both performance and efficiency. Larger variants (B1 to B7) are derived by uniformly scaling the baseline model using the compound scaling principle. This results in models that achieve state-of-the-art performance on image classification tasks with dramatically fewer parameters and lower computational cost compared to earlier architectures like ResNet or Inception.

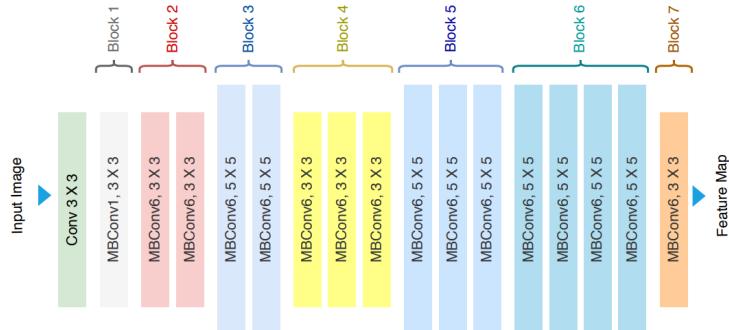


Figure 2.13: Architecture of EfficientNet-B0 with MBConv as Basic building blocks [2].

Lightweight and Specialized CNN Models

Several CNN architectures, such as MobileNet, NASNet, SqueezeNet, and ShuffleNet, have been specifically developed to meet constraints related to speed, model size, and power efficiency, making them suitable for deployment on mobile or edge devices. Although not explored in depth within the main text, a comparative overview of their architectures, key characteristics, and potential applications in smart agriculture is presented in 3.11 for further reference.

2.6 Transfer Learning and Pretrained Models

In deep learning, training models from scratch often requires large labeled datasets and significant computational resources. Transfer learning offers a powerful alternative by leveraging models pre-trained on large benchmark datasets such as **ImageNet**. These pre-trained models capture rich and generalizable features in their initial layers, which can then be adapted to new, often smaller, target datasets with minimal additional training. This section explores the concept of transfer learning, introduces popular pre-trained CNN architectures relevant to agricultural applications, and outlines fine-tuning strategies suitable for small-scale plant datasets.

2.6.1 Concept of Transfer Learning

Transfer learning is a machine learning technique where a model trained on one task is repurposed for a different but related task. In the context of deep learning, it typically involves taking a neural network pre-trained on a large dataset such as ImageNet, which contains over 14 million labeled images, and adapting it to a specific task that may lack sufficient labeled data.

To formalize this, consider a target learning task T_t based on a domain D_t ; transfer learning allows for assistance from a different domain D_s for the learning task T_s . The goal of transfer learning is to improve the performance of the predictive function $f_{T_t}(\cdot)$ for the task T_t by discovering and transferring latent knowledge from D_s and T_s , where generally $D_s = D_t$ and/or $T_s = T_t$. Furthermore, it is often the case that the size of D_s is much larger than that of D_t [53].

This process is illustrated in Figure 2.14, which demonstrates how knowledge from a source task and domain can be transferred to a target task with limited data.

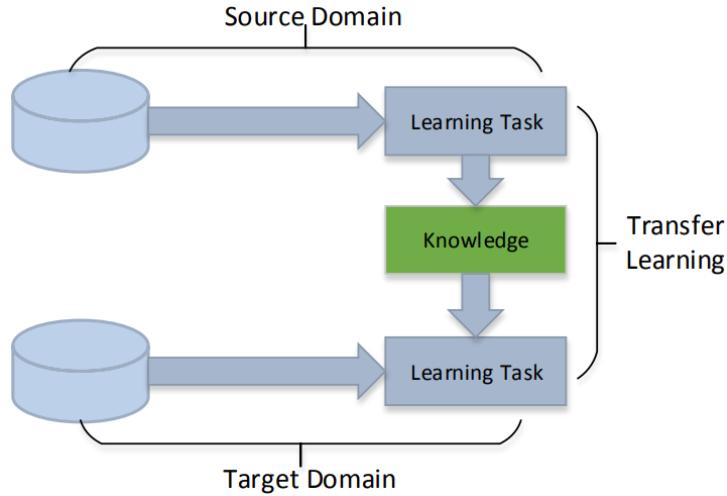


Figure 2.14: Learning process of transfer learning [53].

The two primary approaches to transfer learning are [53]:

- **Feature Extraction:** The pre-trained model is used as a fixed feature extractor. All convolutional layers are kept frozen, and only the final fully connected layer(s) are trained on the new dataset.
- **Fine-Tuning:** Some layers of the pretrained model are unfrozen and retrained on the new dataset. This allows the model to slightly adjust its learned features to better suit the new domain.

This concept is illustrated in Figure 2.15, which demonstrates the integration of a custom CNN with transfer learning networks.

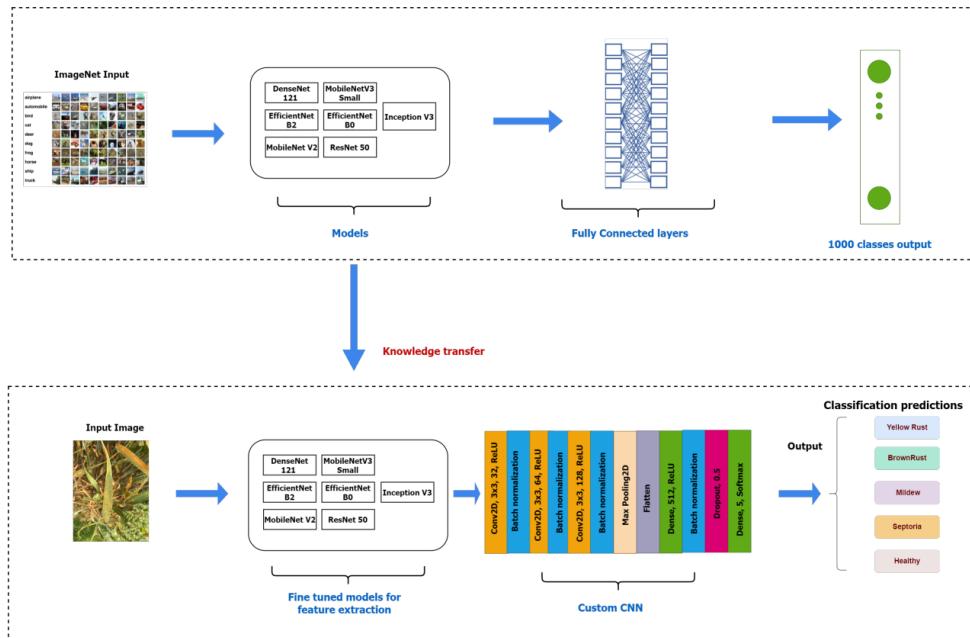


Figure 2.15: Integration of the custom CNN with transfer learning networks [26].

Transfer learning is especially valuable in agriculture, where acquiring large annotated datasets is difficult. By leveraging pre-trained models, researchers and practitioners can build effective models for plant disease detection with limited data and reduced computational cost.

2.6.2 Fine-Tuning Strategies for Agricultural Data

While basic fine-tuning involves unfreezing and retraining a subset of layers in a pretrained model, fine-tuning strategies can be further optimized when dealing with agricultural data, which often presents unique challenges such as class imbalance, limited samples, and high intra-class variability (e.g., similar symptoms across different plant diseases).

In this context, several fine-tuning strategies can be applied to improve model generalization and performance:

- **Gradual Unfreezing:** Instead of unfreezing all layers at once, layers are incrementally unfrozen, starting from the top (fully connected layers) and moving backward. This helps prevent the model from losing previously learned useful features during early training stages [23].
- **Discriminative Learning Rates:** Assigning different learning rates to different layers, lower for earlier layers and higher for later ones, ensures that the more generic features are preserved while higher-level representations are fine-tuned for the new task [23].
- **Early Stopping and Regularization:** Due to limited data, it's essential to prevent overfitting during fine-tuning. Techniques like early stopping, dropout, and weight decay can help maintain model robustness [34].
- **Data Augmentation and Balancing:** Supplementing fine-tuning with targeted data augmentation (e.g., rotation, brightness adjustments, zoom) helps the model generalize better. In addition, synthetic oversampling techniques like SMOTE can address class imbalance issues [5].

These strategies, when integrated thoughtfully, allow researchers to tailor fine-tuning to the specific nature of agricultural datasets, maximizing the benefits of transfer learning even in data-constrained environments.

2.7 Object Detection in Smart Agriculture

In the context of precision agriculture, object detection has emerged as a critical computer vision technique for automating the assessment of crop health. It facilitates the identification and localization of plant diseases, insect pests, nutrient deficiencies, and weeds, supporting more efficient and timely interventions across large-scale farming environments. This approach goes beyond simple image classification by offering spatial

information about multiple objects of interest within a single image, enabling actionable insights for decision-making.

2.7.1 Key Concepts in Object Detection

Understanding object detection requires familiarity with its key components, including the problem definition, how annotated data is structured, and the metrics used to evaluate model performance.

Definition of Object Detection

At its core, object detection involves predicting both the category and precise location of objects within an image, thus combining classification with localization. Traditional approaches consist of stages such as region proposal, feature extraction, and object classification. Over time, detection methods have evolved to address increasing demands for accuracy and speed, giving rise to both two-stage and one-stage detection frameworks [60].

Here are more details about the concepts of object detection [60]:

- **Informative Region Selection:** It is used to identify specific areas within an image where objects are likely to appear. This step helps reduce computational load by focusing only on promising regions instead of scanning the entire image at all scales and positions. In agricultural imagery, objects like plants or pests may vary in size, shape, and location. Early methods used multiscale sliding windows to generate candidate regions, but this approach was computationally expensive and often produced redundant proposals. To improve efficiency, modern techniques now use region proposal algorithms or attention mechanisms to better focus on meaningful areas while avoiding irrelevant ones.
- **Feature Extraction:** It is the process of identifying and isolating relevant visual attributes from an image that can effectively represent objects within it. Object recognition involves detecting characteristics that are both robust and semantically significant. Traditional methods such as Scale-Invariant Feature Transform (SIFT), Histograms of Oriented Gradients (HOG), and Haar-like features were designed to mimic human vision by emphasizing edges, textures, and patterns. However, these handcrafted techniques often struggled to maintain performance due to challenges like changes in object appearance, lighting variations, and cluttered backgrounds, leading to their limitations in complex scenarios.
- **Classification and Localization:** Classification and localization refer to the process of both identifying the object in an image and determining its precise location through bounding boxes. With the rise of deep learning, this step underwent a significant transformation. Models such as R-CNN and its subsequent versions—Fast R-CNN, Faster R-CNN, and YOLO—automated the feature extraction process and

seamlessly integrated classification with bounding box regression. These advancements led to substantial improvements in both detection accuracy and processing speed, enabling real-time applications in fields like crop monitoring and pest detection.

Annotation of Objects

Annotation refers to the process of labeling the objects (such as pests, diseases, or damaged crops) in the images by drawing bounding boxes around them and assigning class labels. This is a critical step for supervised learning, where the model learns to identify patterns based on labeled data. Several annotation techniques can be used:

- **Bounding Boxes:** The most common annotation method in object detection. Each object is enclosed in a rectangular box, and the class of the object is assigned to it (e.g., "rust", "aphid").
- **Polygons:** For more precise object delineation, especially when objects have irregular shapes (e.g., plant leaves affected by disease), polygons are used instead of bounding boxes.
- **Semantic Segmentation:** In cases where the task involves classifying each pixel in the image, semantic segmentation labels each pixel to indicate which class it belongs to (e.g., diseased or healthy tissue in a leaf).

Evaluation Metrics in Object Detection

In object detection, several metrics are used to assess model performance:

- **Mean Average Precision (mAP):** Measures the average precision across all object classes, balancing precision and recall to evaluate overall model performance.
- **Intersection over Union (IoU):** Calculates the overlap between predicted and ground truth bounding boxes, indicating localization accuracy. Higher IoU means better localization.
- **Precision and Recall:**
 - **Precision:** Measures the proportion of true positive detections out of all predicted objects.
 - **Recall:** Measures the proportion of true positive detections out of all actual objects.
- **F1-Score:** The harmonic mean of precision and recall, providing a balance between both.
- **Average Recall (AR):** Evaluates recall at different IoU thresholds, useful for detecting small objects or handling occlusions.

- **Confusion Matrix:** Summarizes true positives, false positives, true negatives, and false negatives, providing insights into model errors.
- **Speed Metrics (FPS, Latency):** Assess real-time performance, essential for time-sensitive applications like precision agriculture.
- **AP at Specific IoU Thresholds:** Measures precision at different IoU levels to understand performance under stricter conditions.

2.7.2 Key Architectures

Many architectures are designed to efficiently and accurately detect objects in images, even in complex agricultural environments. Below, we discuss four of the most widely used and effective object detection models: YOLO (You Only Look Once), R-CNN, Faster R-CNN, and SSD (Single Shot Multibox Detector).

Yolo

YOLO is a fast and efficient object detection framework that predicts both object confidences and bounding boxes (BBs) using the entire topmost feature map. The image is divided into a $S \times S$ grid, where each grid cell is responsible for predicting objects centered within it. Each cell predicts multiple bounding boxes and their corresponding confidence scores, which reflect the likelihood of an object being present and how well the predicted box overlaps with the ground truth (IoU) (see figure 2.16).

At test time, class-specific confidence scores are computed by multiplying the box confidence with conditional class probabilities. YOLO optimizes a loss function during training to fine-tune predictions and improve detection accuracy [60].

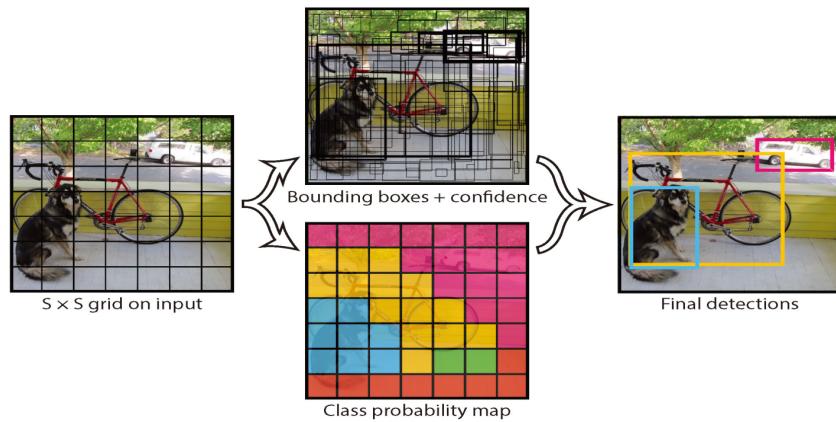


Figure 2.16: Main idea of YOLO [60].

R-CNN

R-CNN is a significant advancement in object detection, improving the quality of candidate bounding boxes (BBs) and utilizing deep architecture for high-level feature extraction [60]. It consists of three main stages, as presented in figure 2.17:

0. **Region Proposal Generation:** R-CNN uses selective search to generate about 2000 region proposals per image, improving candidate box accuracy and reducing the search space.
0. **CNN-Based Feature Extraction:** Each region proposal is resized and passed through a CNN to extract a 4096-dimensional feature, creating a high-level, robust representation of the object.
0. **Classification and Localization:** Region proposals are classified using pre-trained linear SVMs, and bounding box regression is applied. Non-maximum suppression (NMS) is used to eliminate redundant boxes and finalize object detections.

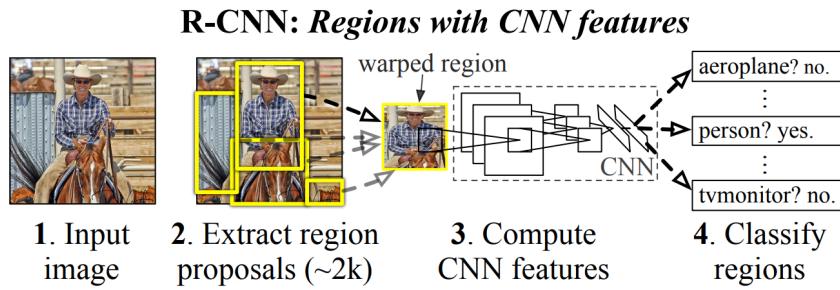


Figure 2.17: Flowchart of R-CNN [60].

Despite its success, R-CNN has drawbacks, including slow inference due to CNN computation for each region, time-consuming multistage training, high memory and storage requirements for storing region features, and redundant region proposals from selective search that slow down the process [60].

Faster R-CNN

Faster R-CNN improves upon earlier object detection models by introducing a Region Proposal Network (RPN), a deep learning-based method for generating object proposals, which shares convolutional features with the detection network to generate object proposals efficiently, eliminating the need for methods like selective search. The RPN uses a fully convolutional network (FCN) (as presented in Figure 2.18) to predict bounding boxes and object scores simultaneously. The system uses anchors of multiple scales and aspect ratios and is trained end-to-end with a multitask loss function. While Faster R-CNN achieves state-of-the-art accuracy and high-speed processing, it is limited by its alternate training algorithm, which is time-consuming and struggles with extreme object scales and shapes [60].

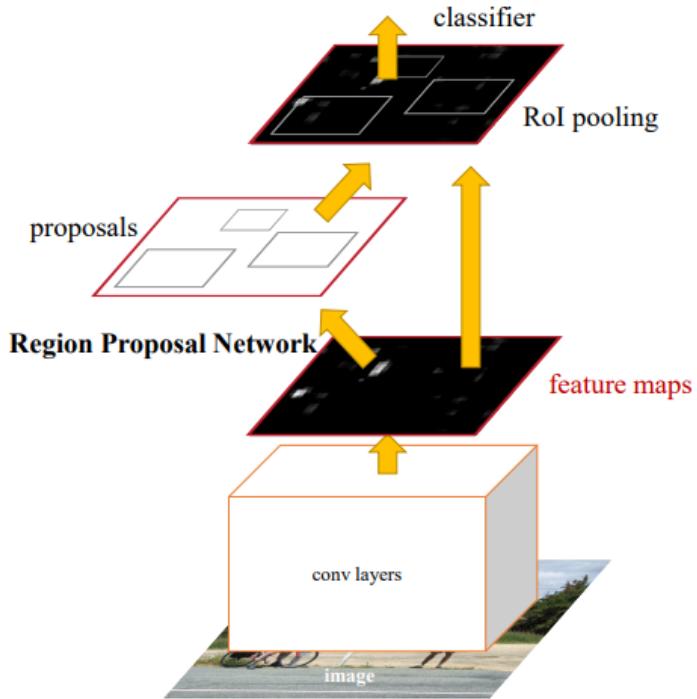


Figure 2.18: An illustration of the Faster R-CNN model [46].

SSD

SSD was introduced to address the limitations of YOLO, particularly in handling small objects and objects with unusual aspect ratios. Unlike YOLO's fixed grid approach, SSD uses default anchor boxes of various aspect ratios and scales to better handle objects of different sizes.

It integrates predictions from multiple feature maps with different resolutions and uses a VGG16 backbone architecture with additional layers for bounding box predictions. SSD is trained with a combination of localization and confidence losses and refines detections using non-maximum suppression (NMS). It outperforms Faster R-CNN in accuracy on PASCAL VOC and COCO while being three times faster, running at 59 fps with an input size of 300×300 . However, SSD still struggles with small objects, which can be improved with better feature extractors and network modifications [60].

Below is the architecture of SSD (Figure 2.19), illustrating its key components, including the VGG-16 backbone, extra feature layers, and classifier convolutions.

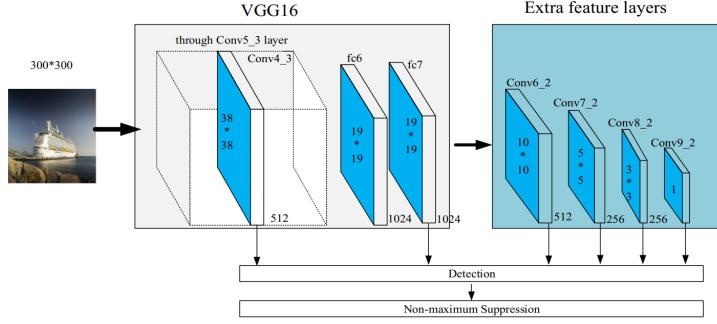


Figure 2.19: Architecture of SSD [33].

2.8 Challenges of Deep Learning in Agricultural Contexts

Despite the promising performance of deep learning and object detection in various fields, their application in agriculture presents a range of specific challenges. One of the primary issues is the limited availability of large, annotated agricultural datasets, which hampers the training of robust and generalizable models. Unlike natural image datasets like ImageNet, agricultural datasets often suffer from class imbalance, incomplete labeling, and domain specificity (e.g., crop types, diseases, and environmental conditions) [4].

Another challenge is the high intra-class variability and low inter-class variability found in agricultural images. For instance, symptoms of different diseases may look visually similar, or the same disease may appear differently across plant species and growth stages. Additionally, variations in lighting, occlusion by leaves, overlapping plants, and background clutter significantly affect detection performance [4].

The seasonal and geographic diversity further complicates the generalization of models trained on limited datasets. Moreover, the deployment of deep learning models in real agricultural environments must consider limited computing resources, especially for edge or mobile devices used in fields.

Lastly, ensuring the interpretability and trustworthiness of AI decisions is crucial in agricultural contexts, as these technologies directly impact yield, resource use, and farmers' livelihoods. Addressing these challenges requires collaborative efforts in data collection, annotation, model adaptation, and efficient deployment strategies [4].

2.9 Conclusion

In conclusion, this chapter explored the integration of deep learning techniques, especially convolutional neural networks (CNNs), into agricultural contexts, focusing on their role in plant disease classification and object detection. We introduced key CNN architectures such as VGGNet, ResNet, DenseNet, and EfficientNet, each contributing unique design principles and performance improvements. Transfer learning was also discussed as an

effective strategy to overcome data scarcity by adapting pretrained models to agricultural datasets with minimal resources.

We then highlighted the concept of object detection and its significance in precision agriculture for tasks like identifying pests, diseases, and weeds. Core ideas such as region proposal, feature extraction, and classification were presented, along with modern deep learning-based solutions. Finally, the chapter addressed real-world challenges, such as image variability, environmental noise, and limited annotated data, all of which must be considered when developing robust AI-based agricultural systems. In the following chapter, we will explore hybrid approaches, such as ensemble learning and its integration with CNN models, to further enhance accuracy, generalization, and robustness in agricultural applications.

Chapter 3

Integrating Remote Sensing and AI-Based Methods for Detecting Wheat Diseases

3.1 Introduction

Remote sensing technologies play a crucial role in modern agriculture by enabling large-scale and timely monitoring of crops. Through satellite and drone imagery, farmers can observe wheat fields, assess plant health, and detect early signs of disease. These technologies provide valuable data that, when combined with machine learning (ML) and deep learning (DL) techniques, allow for accurate identification and classification of wheat diseases.

This chapter presents an overview of the imaging tools and platforms used in agricultural remote sensing, followed by a focus on how ML/DL methods process this data to detect diseases in wheat crops. It highlights key workflows, data fusion techniques, and real-world examples, offering a practical perspective on improving disease management through technology.

3.2 Imaging Technologies in Remote Sensing

Imaging technologies play a key role in remote sensing systems by capturing high-resolution images of the Earth's surface, which are crucial for agricultural monitoring. These technologies use different types of cameras to detect various parts of the electromagnetic spectrum (Figure 3.1), enabling detailed analysis of vegetation, soil, and environmental conditions. The choice of camera depends on the specific agricultural purpose, such as disease detection, crop monitoring, or environmental assessment.

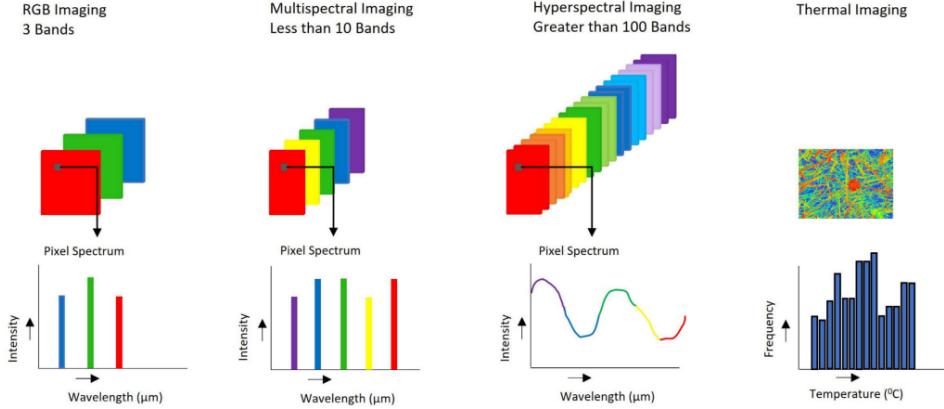


Figure 3.1: Image acquisition techniques [19].

3.2.1 RGB Cameras (Red, Green, Blue)

RGB cameras are the most common type of imaging technology used in remote sensing. They capture images in the visible light spectrum (red, green, and blue wavelengths), similar to how the human eye perceives the world [10]. As shown in Figure 3.2, RGB imagery can also be processed to remove non-vegetative elements such as soil background, enhancing the visibility of plant features [35].

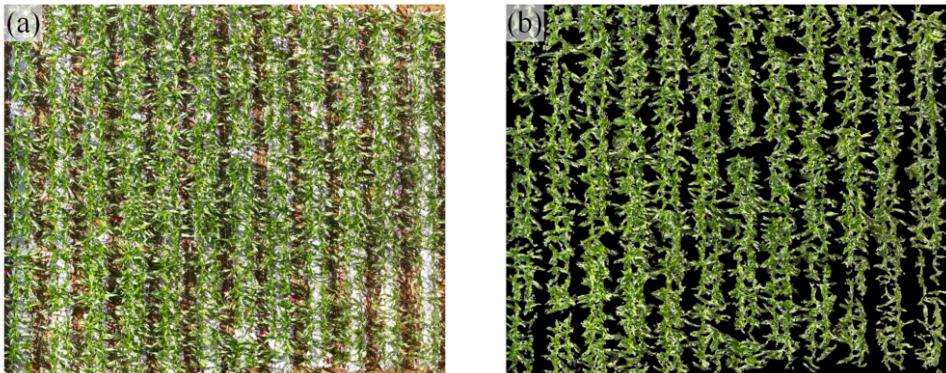


Figure 3.2: RGB original image (a) and RGB image after removal of soil background (b) [35].

3.2.2 Near-Infrared (NIR) Cameras

Near-infrared (NIR) cameras capture images in the near-infrared spectrum, which lies just beyond the visible light range. These cameras are sensitive to light wavelengths that are not visible to the human eye, typically ranging from 700 to 1300 nm [10]. NIR imaging is particularly valuable in plant health monitoring as it can detect subtle changes in vegetation that are not visible in the visible light spectrum. Figure 3.3 shows an example of a UAV equipped with NIR sensors used for such applications.



Figure 3.3: The UAV and its sensors [35].

3.2.3 Thermal Infrared Cameras

Thermal infrared cameras capture images based on the heat emitted by objects, operating in the infrared spectrum (wavelengths typically ranging from 8,000 to 14,000 nm). They detect temperature differences and translate them into visual representations, making them invaluable for identifying heat-related patterns in crops and soil [10]. This capability is demonstrated in Figure 3.4, which shows thermal and RGB images captured at different UAV flight heights.

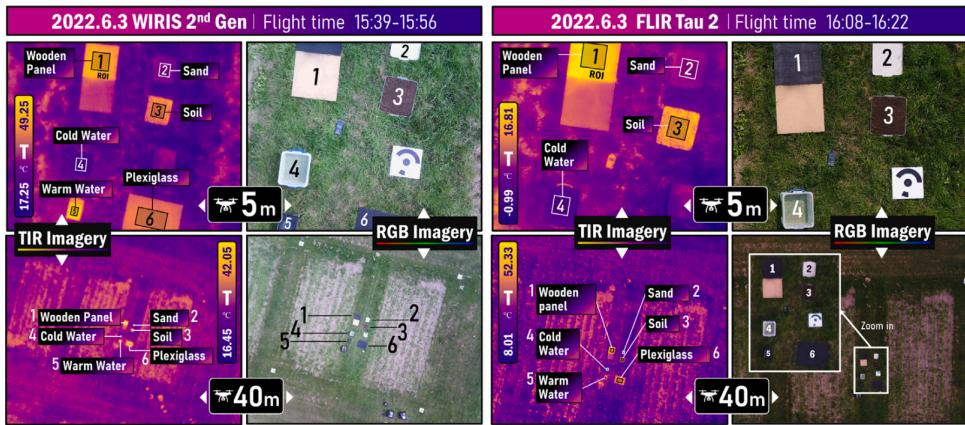


Figure 3.4: Examples of acquired thermal images from FLIR Tau 2 and WIRIS 2nd Gen at two different UAV flight heights. RGB images acquired by the RGB cameras mounted on the same UAVs were also provided. “T” in the graphs refers to the measured temperature by the corresponding thermal cameras [54].

3.2.4 Multispectral Cameras

Multispectral cameras capture images across multiple spectral bands, including visible (red, green, blue) and non-visible wavelengths (near-infrared and red-edge) across a lim-

ited number of discrete bands, each covering a wider spectral range spanning tens to hundreds of nanometers [38]. These cameras typically have 5 to 10 bands, enabling detailed analysis of plant health and environmental conditions [10]. They are essential for applications that require data beyond what RGB cameras can provide.

3.2.5 LiDAR Cameras (Light Detection and Ranging)

LiDAR cameras use laser pulses to measure distances by calculating the time it takes for the laser to bounce back from a surface. This technology generates high-resolution 3D maps, capturing detailed information about the shape and structure of the terrain, vegetation, and other objects. LiDAR operates effectively in various environmental conditions, including low light or dense canopy cover [10].

3.2.6 Hyperspectral Cameras

Hyperspectral cameras capture images across a vast range of continuous, narrow spectral bands, often exceeding hundreds. These bands typically have a spectral resolution below 10 nanometers, covering both visible and non-visible wavelengths. The fine-scale spectral data collected allows for the detection of subtle variations in reflectance, enabling detailed analysis of crop conditions, such as disease identification and nutrient stress. This high-resolution imaging is crucial for precision agriculture, where detecting subtle differences can significantly impact decision-making and yield optimization [10] [38].

3.3 Remote Sensing Platforms

Remote sensing platforms are categorized based on their altitude and mobility, each offering unique capabilities for capturing agricultural data. The three main types are satellite-based imaging, UAV (drone) imaging, and aircraft imaging.

3.3.1 Satellite-Based Imaging

Satellite-based imaging involves the use of Earth-observing satellites equipped with multispectral sensors to capture detailed images of the Earth's surface. These satellites, such as Quickbird and Landsat-1, can capture high-resolution images, providing data with varying levels of precision. Early satellite systems offered up to 6.5 meters per pixel in resolution [44], while modern systems have significantly improved both spatial and spectral resolution. These advancements enable more accurate and comprehensive monitoring of large areas, laying the foundation for a wide range of applications, including agriculture and environmental studies [1]. Despite its many advantages, satellite-based imaging also presents several challenges and limitations that must be considered in agricultural applications, including:

- **Coarse Resolution:** The resolution of satellite imagery can be too coarse for some applications. For instance, imagery from platforms like Sentinel-2 (up to 10 m resolution) may not be detailed enough for fields with closely spaced rows, such as vines, leading to mixed pixel data that combines multiple rows and soil [1].
- **Data Processing Complexity:** Due to the coarse resolution, advanced methods like computer vision classifiers and statistical decision trees are often required to extract useful information, such as detecting shrubs or distinguishing plant species [phang2023from].
- **Mixed Pixel Issue:** Low-resolution pixels, like those from Landsat, result in mixed data, complicating analysis and interpretation, especially in detailed applications [10].
- **Spatiotemporal Challenges:** Obtaining timely spatiotemporal data on crop phenological status during critical growth periods is difficult, especially due to cloud coverage [10].
- **High Costs:** Accessing satellite data equipped with multispectral sensors can be expensive, which limits its use for some applications [10].

3.3.2 Aircraft-Based Imaging

Aircraft-based imaging involves the use of piloted or unpiloted aircraft, such as airplanes, equipped with various remote sensing tools to capture high-resolution imagery and sensor data over large agricultural areas. These systems offer a significant advantage in large-scale applications due to their flexibility and ability to carry heavier payloads of sensors, making them a viable alternative to satellite-based or UAV-based solutions [10]. Before the widespread adoption of UAVs, manned aircraft were commonly employed for lower-to-ground remote sensing, utilizing multi-spectral or electro-optic (EO) sensors to monitor agricultural conditions [44]. However, this approach also faces several limitations, including:

- **High Operational Costs:** Operating aircraft-based systems involves significant expenses for fuel, maintenance, and pilot salaries, making them less affordable compared to UAVs [phang2023from, 10].
- **Complex Logistics and Pilot Requirement:** Certified pilots and specific flight logistics are essential, adding complexity and reducing flexibility in deployment [1].
- **Limited Flexibility:** Aircraft require designated takeoff and landing zones, restricting their use in remote or uneven terrains [10].
- **Weather Sensitivity:** Adverse weather conditions, such as strong winds or rain, can impact flight stability and data quality, limiting operations during critical periods [10].
- **Regulatory and Airspace Restrictions:** Aircraft-based systems are subject to strict aviation regulations and airspace restrictions, limiting their operational scope [10].

- **High Data Processing Requirements:** Large volumes of data generated require specialized software and computing resources, making data processing resource-intensive [10].
- **Costly for Large-Scale Use:** The expense and complexity of these systems make them impractical for frequent large-scale monitoring, favoring UAV alternatives for cost efficiency [phang2023from].
- **Resolution Limitations:** Although aircraft provide better resolution than satellites, their imagery can still be too coarse for some precision applications [1].

3.3.3 UAV-Based Imaging

Unmanned Aerial Vehicles (UAVs), or drones, are versatile remote sensing platforms that have become essential tools in remote sensing due to their cost-effectiveness, flexibility, and ability to capture high-resolution (cm-level) images, making them ideal for precision agriculture [51]. UAV-based imaging typically involves using low-altitude remote Sensing Systems (LARS) to acquire detailed imagery of the Earth's surface at low altitudes, providing high precision and adaptability [44].

Unlike traditional satellite platforms, UAVs offer several advantages, including on-demand data collection, high-resolution imagery, and flexible deployment, which enable real-time monitoring and analysis [44]. A complete Unmanned Aerial System (UAS) includes the UAV and its remote sensing equipment, operating without a human pilot onboard, and is capable of carrying various sensors tailored to different agricultural needs [1]. Despite their advantages, UAVs also face some challenges, including:

- **Short Flight Duration:** UAVs, especially smaller models, often have flight durations of less than 30 minutes, which limits their coverage area, particularly for large-scale agricultural operations [phang2023from].
- **Regulatory Challenges:** Stricter regulations, especially for larger UAVs, slow down their adoption and innovation, hindering their widespread use [phang2023from, 10].
- **Scalability:** UAV-based remote sensing requires trained pilots and continuous monitoring, which limits scalability, particularly for small-scale applications [phang2023from].
- **Environmental Sensitivity and Calibration Issues:** Hyperspectral sensors on UAVs face challenges related to environmental factors such as light exposure and atmospheric interference, necessitating frequent recalibration [1].
- **Weather Dependency:** UAVs are susceptible to weather conditions like wind and rain, which can affect both flight stability and data quality [10].
- **Battery Life:** The limited battery life of UAVs impacts their ability to cover large areas, particularly in extensive agricultural fields [10].

3.4 Vegetation Indices

Vegetation indices are mathematical formulas that combine light reflectance data from different spectral bands (such as visible, near-infrared, and mid-infrared) captured by sensors on drones or satellites to assess plant health and condition. These indices provide valuable insights into growth stages, plant vigor, biomass, and chlorophyll levels [51]. Plants reflect sunlight differently based on their type, structure, and water content: they reflect little in the blue and red regions, more in green (which makes them appear green), and strongly in the near-infrared (NIR) if healthy (Figure 3.5).

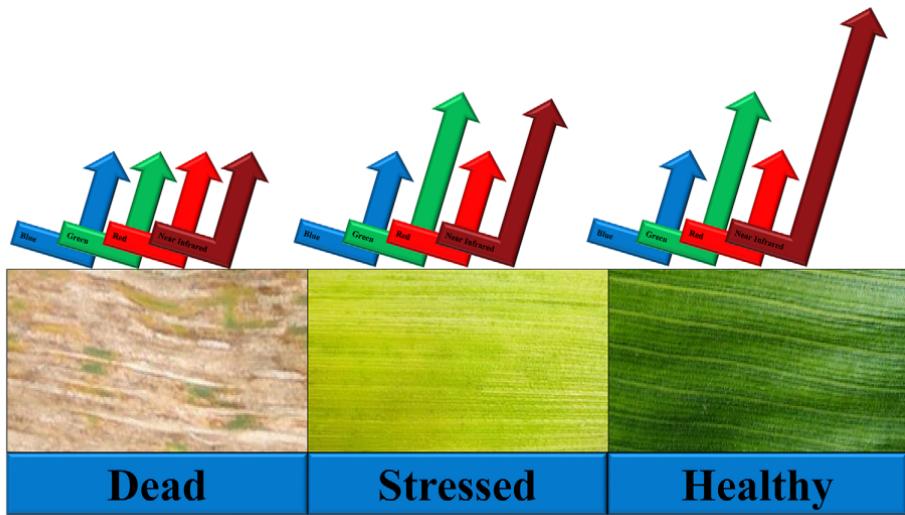


Figure 3.5: Unique optical reflectance signature differences of blue, green, red, and near-infrared light emitted from dead, stressed, and healthy plant tissue [42].

3.5 Image processing

Imagery processing is the technique used to turn multiple aerial images captured by drones (UAVs) or remote sensors into a single, accurate, and clear image known as an orthomosaic. Since drones don't capture one large image but instead take many smaller ones of different parts of the area, these images must be merged using a process called image stitching (as shown in Figure 3.6). This involves merging individual images into one large composite by identifying and aligning "key points" such as a rock, plant, or edge of a field. While the drone captures these images, it simultaneously records GPS or location metadata, which is then used in geographic alignment to position each image on a map accurately. The final output of this process is known as data products, which can include color mosaics (stitched colored images), spectral mosaics (capturing invisible wavelengths like near-infrared for agricultural analysis), thermal mosaics (highlighting temperature variations), surface and terrain models (representing elevation and landscape shapes), and point clouds (dense 3D representations of surfaces for detailed analysis) [42].

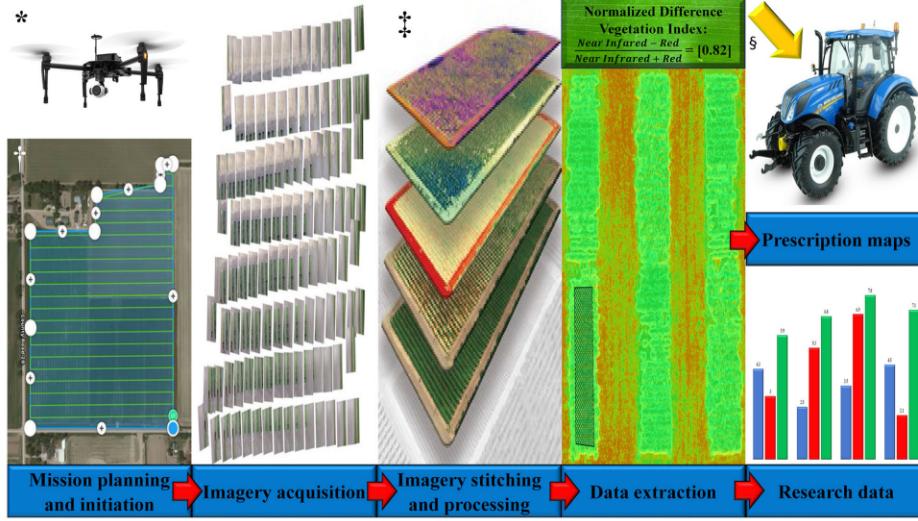


Figure 3.6: Standard Workflow for Generating UAV-Derived Imagery and Data Products [42].

Imagery can be processed on-site with personal computers and the appropriate software or in the field with cloud computing imagery processing [42]. The pros and cons of each imagery processing option are discussed in Table 3.1.

	Pros	Cons
Software (On-Site Processing on PC)	Customizable options	High-performance PC needed
	No limits on images	Expensive initial cost
	GIS software compatible	Steep learning curve
	One-time purchase	Slower processing
Cloud Computing (Online/Re- mote Processing)	Easy to use	Fewer customizations
	Low upfront cost	Image limits
	Fast field processing	Ongoing costs
	No powerful PC needed	
	Simplified analysis	

Table 3.1: Comparison of On-Site Software vs. Cloud Computing for Imagery Processing [42].

3.6 Applications of Remote Sensing Data

Remote sensing has diverse applications in agriculture, enabling precise, data-driven decision-making. It helps monitor field conditions, optimize resource use, and address

issues early. The following sections highlight key areas where remote sensing enhances farm management:

- **Crop Health Monitoring:** Remote sensing enables regular monitoring of crop conditions, helping detect diseases, pest presence, and water or nutrient stress early. Vegetation indices (e.g., NDVI) derived from RGB, NIR, multi-spectral, and hyperspectral sensors are commonly used to assess aspects such as foliage cover, pigment composition, and water stress. LiDAR and GPS can also support the accurate mapping of plant structure and spatial positioning. These technologies collectively support better crop management and yield optimization [44].
- **Weed Control:** Precise weed detection helps reduce competition for water, nutrients, and light. Aerial imagery from UAVs and satellite platforms, often using RGB and multispectral cameras, supports vegetation indexing and mapping for weed identification. Some systems enable real-time onboard image analysis and spraying. These tools assist in implementing targeted and efficient weed management strategies while reducing input costs [44].
- **Infectious Disease Epidemiology and Mapping:** UAVs are used to gather high-resolution spatial data for studying the relationships between environmental conditions and disease spread. They help monitor changes in land use, population distribution, and vegetation patterns, which are valuable for identifying and mitigating disease risks affecting crops or livestock. UAV-collected data provide flexible, cost-effective alternatives to satellite and manned aerial surveys [44].
- **Spectral Imaging:** Multispectral and hyperspectral imaging technologies offer detailed insights into crop nutrient levels, stress, and overall vigor. Though hyperspectral data offer higher spectral resolution, they also require more complex processing, often conducted post-flight. These systems rely on sensors like CCD and CMOS and use scanning methods such as point, line, and plane scanning. Accurate georeferencing is achieved using ground control points (GCPs) or advanced navigation systems [44].

3.7 Workflow for Integrating Remote Sensing and ML/DL in Wheat Disease Detection

Existing research on crop disease detection using UAV imagery can be grouped into three main methodological approaches (Figure 3.7). The first involves statistics-based methods, which apply correlation and regression analyses to identify linear relationships between disease symptoms and spectral data from UAV images. These methods commonly use vegetation indices (VIs) to extract relevant crop traits. The second approach includes conventional machine learning techniques, where VIs serve as input features for supervised or unsupervised models used in disease estimation. The third and most recent approach is based on deep learning, which leverages raw UAV images alongside other features to train end-to-end models for automated crop disease recognition [48].

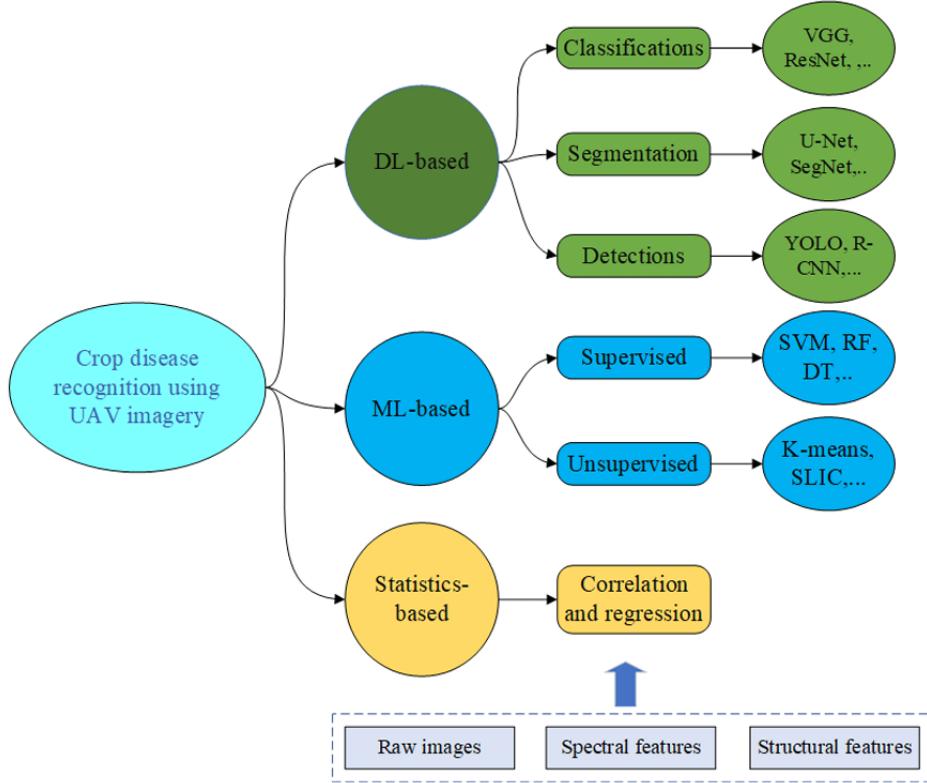


Figure 3.7: Classification of crop disease detection approaches using UAV-based remote sensing. The elements within the dotted box represent the image features utilized across one or more of the listed approaches [48].

3.7.1 Features Extraction

In the context of remote sensing for plant disease detection, different types of features can be extracted from imagery to capture various physiological, structural, and visual properties of crops.

- **Spectral Features:** Describe how plants reflect light across different spectral bands (such as visible, red edge, or near-infrared). These features reveal the biochemical state of vegetation, helping detect conditions like chlorophyll content, moisture level, and early stress symptoms using vegetation indices (e.g., NDVI, NDRE) or raw spectral band reflectance [12].
- **Structural Features:** Represent the physical shape, size, and arrangement of plant components, such as canopy height, leaf area, or plant density. They help monitor growth patterns and identify visible deformations caused by disease, pests, or environmental stress through 3D models or spatial measurements [12].
- **Textural Features (TF):** Capture the variation and arrangement of pixel intensities in an image, describing surface patterns like roughness, smoothness, or regularity. These features are useful for detecting visual symptoms of disease, such as lesions or uneven leaf surfaces, using statistical metrics like contrast, entropy, and homogeneity [12].

- **Wavelet Features (WF):** Capture spatial and spectral variations in data by breaking it down into components at different scales. These features can detect fine-scale patterns, which are useful for identifying subtle changes in vegetation, such as variations caused by disease or stress [39].

3.7.2 Statistics-Based Methods

Statistics-based methods for crop disease estimation rely on using crop-related traits derived from UAV imagery as independent variables and disease scores as the target variable. These methods typically involve three main steps: image pre-processing, vegetation index (VI) generation, and statistical analysis. During pre-processing, spatial data products such as reflectance maps are generated, representing how much light vegetation reflects across different wavelengths, which is crucial for assessing plant health. These maps calculate various vegetation indices (e.g., NDRE, NDVI, and DVI) to summarize plant conditions like chlorophyll content, stress, or canopy vigor. These indices are then used in correlation and regression analyses to estimate disease severity [48].

In practice, fields are often divided into small plots, and the mean VI value per plot is computed for analysis. Some studies also apply threshold-based techniques, setting a cutoff value to distinguish between healthy and diseased areas. However, defining a universal threshold is difficult due to variability in crop type, disease symptoms, and imaging conditions.

For example, [20] employed UAV-based hyperspectral imagery to detect wheat yellow rust and incorporated both vegetation indices (VIs) and texture features (TFs) into their statistical models. [20] extracted TFs using Principal Component Analysis (PCA), followed by methods like the Gray-Level Co-occurrence Matrix (GLCM), a technique used to analyze the spatial relationship between pixel intensities in an image. GLCM captures texture features such as contrast, entropy, and correlation, which help detect subtle patterns in plant surfaces that could indicate disease or stress. By combining these TFs with spectral indices in a Partial Least Squares Regression (PLSR) model, they achieved significantly higher accuracy, reaching an R^2 of 0.88 in the late infection stage. PLSR is a regression technique that is particularly effective when the predictors are many and possibly collinear, as it reduces dimensionality and finds the most relevant directions for predicting the response variable. The general form of the PLSR equation (3.1) is:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n \quad (3.1)$$

where Y is the disease severity index, X_1, \dots, X_n are the input features (e.g., VIs, TFs), and β_0, \dots, β_n are the regression coefficients determined during model training.

Their study also highlighted that high spatial resolution (10–15 cm) images were critical for capturing fine structural differences caused by disease. Similarly, [7] achieved a strong correlation ($R^2 = 0.79$) using the GLI index derived from RGB images to assess foliar diseases in wheat (Table 03). Together, these studies show that combining spectral and textural information enhances the performance of statistical models in UAV-based crop disease monitoring.

The detailed formulas for the vegetation indices (VIs) used in the studies mentioned are provided in Annex A.

Paper	Disease	Sensor	VIs	Eval. Metrics
[7]	Wheat foliage disease	RGB	NDI, GI, and GLI	$R^2 = 0.79$
[22]	Leaf rust, Stem rust	RGB	SRI, LRI	$R^2 = 0.81$
[20]	Yellow rust	Hyperspectral	SIPI, PRI, TCARI, PSRI, YRIGI	$R^2 = 0.88$
[30]	Powdery Mildew	Hyperspectral	PMI, MSR, MCARI	$R^2 = 0.722$

Table 3.2: Summary of ST-based methods for Wheat disease estimation using UAV imagery.

3.7.3 Conventional Machine Learning (ML)-Based Methods

Traditional machine learning (ML) methods, including support vector machines (SVMs), artificial neural networks (ANNs), and random forests (RFs), have been applied to crop disease detection using UAV imagery. These methods aim to identify patterns in labeled or unlabeled data, with supervised learning relying on labeled datasets for training and unsupervised learning exploring hidden patterns without labels [48]. The typical ML pipeline involves data collection, preprocessing, feature extraction, and model building, as shown in Figure 3.8.

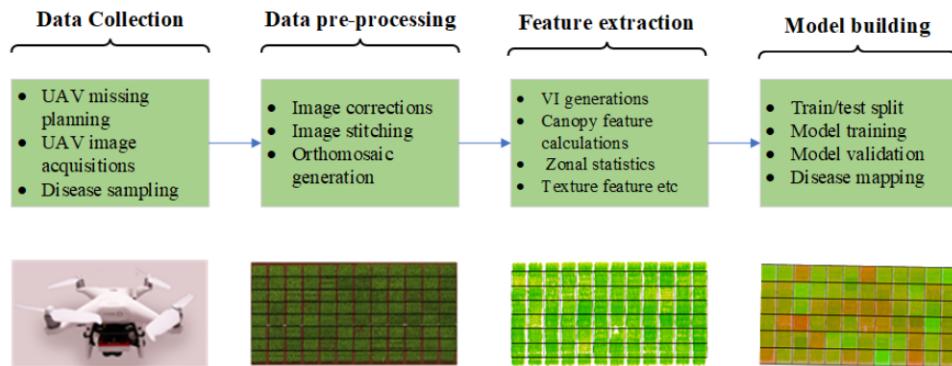


Figure 3.8: General Workflow of Conventional ML-Based Crop Disease Detection Using UAV Imagery [48].

Researchers have applied various machine learning (ML) methods to detect wheat diseases, with different results depending on the disease type and the input features used. Here is a detailed explanation of the studies:

- [37] worked on detecting Fusarium head blight using hyperspectral sensors. They utilized spectral bands, vegetation indices, and texture features as input features. They applied a Backpropagation neural network with Simulated Annealing, a technique that helps optimize the model by avoiding local minima, and achieved a high accuracy of 98%.

- [39] focused on detecting Fusarium head blight using hyperspectral sensors. They used spectral bands, vegetation indices, and wavelet features. The machine learning method employed was Support Vector Machine (SVM), which yielded a coefficient of determination (R^2) of 0.88, indicating a strong predictive performance.
- [61] worked on detecting wheat scab using multispectral sensors. They used vegetation indices and texture features as input features. The models used for analysis included Partial Least Squares Regression (PLSR), Support Vector Regression (SVR), and Backpropagation Neural Networks (BPNN), achieving an R^2 value of 0.83.
- [8] focused on detecting yellow rust using hyperspectral sensors, with vegetation indices as the primary input feature. They applied the Support Vector Machine (SVM) method and obtained an R^2 value of 0.63.

Table 3.3 provides a summary of conventional machine learning methods for wheat disease estimation, highlighting the combination of sensor types, features, and machine learning techniques employed to improve disease detection accuracy.

Reference	Disease	Sensors	Features	ML Methods	Eval. Metrics
[37]	Fusarium head blight	HS	SBs (spectral bands), VI, and TF (texture features)	BP with SA	Accuracy = 0.98
[39]	Fusarium head blight	HS	SBs (spectral bands), VIs, and WFs	SVM	$R^2 = 0.88$
[61]	Wheat scab	MS	VI and TF (texture features)	PLSR, SVR, and BPNN	$R^2 = 0.83$
[8]	Yellow rust	HS	VIs	SVM	$R^2 = 0.63$

Table 3.3: Summary of conventional ML methods for wheat disease estimation.

3.7.4 Deep Learning (DL)-Based Methods

Deep learning methods have been widely applied for crop disease estimation using UAV imagery. The general pipeline for deep learning-based crop disease detection is shown in Figure 3.9 and includes data collection, preparation, model building, and evaluation. However, specific tasks such as image stitching, tiling, and annotation are critical during data preparation for UAV images. Deep learning models for crop disease detection can be categorized into classification-based, segmentation-based, and detection-based approaches. Segmentation models classify individual pixels as healthy or diseased, while classification models classify entire images into disease categories [48]. Detection-based models localize and label the area of interest within the image, as illustrated in Figure 3.9.

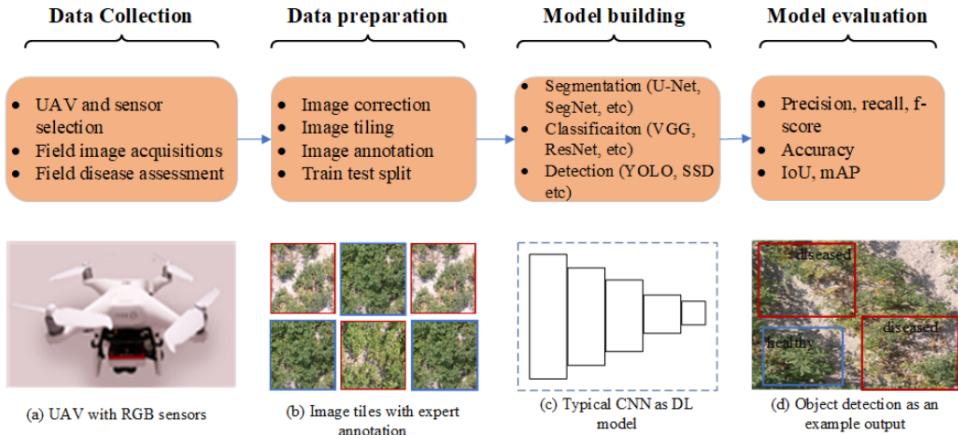


Figure 3.9: General Workflow of DL-based crop disease detection using UAV imagery [48].

Pixel-Based Segmentation Models

Pixel-based segmentation models focus on classifying each pixel in an image based on its characteristics, allowing for precise detection of disease symptoms in specific areas of a plant or field. By analyzing the image at a granular level, these models assign labels to each pixel, helping to identify localized disease patterns across the crop. Several studies have applied such models to detect wheat diseases, yielding impressive results. For instance, [43] used RGB sensors and the PSPNet model to detect yellow rust, achieving an accuracy of 94%. Similarly, [52] applied multispectral sensors and the U-Net model for yellow rust, achieving a recall of 0.926 and an F-score of 0.92. [11] employed RGB sensors with the DeepLabv3+ model for stem rust, obtaining an F-score of 0.81. [56] used RGB sensors and the lr-UNet model for yellow rust, achieving an accuracy of 97.13%, while [57] applied multispectral sensors with the UNet and DF-UNet models for yellow rust, achieving an accuracy of 96.93%. These studies demonstrate the effectiveness of pixel-based segmentation models in detecting wheat diseases with high precision.

Object-Level Classification Models

Object-level classification models treat the image as a whole, classifying entire regions or objects within an image rather than individual pixels. These models focus on detecting the overall presence of diseases at the object or plant level, which is useful for broader assessments of crop health. For instance, [58] used hyperspectral sensors and the Inception-ResNet model for detecting yellow rust, achieving an accuracy of 85%. Similarly, [24] applied RGB sensors and a convolutional neural network (CNN) to detect Helminthosporium leaf blotch, achieving an accuracy of 91.43%. These studies highlight the application of object-level classification models in detecting crop diseases at a larger scale, emphasizing overall disease presence across plant regions.

A summary of deep learning-based methods for wheat disease estimation is presented in Table 3.4. Overall, these deep learning approaches show great potential for fine-grained and scalable wheat disease monitoring in precision agriculture.

Reference	Disease	Sensors	Task	DL Methods	Results
[43]	Yellow Rust	RGB	Pixel-Based Segmentation Models	PSPNet	Acc = 0.94
[52]	Yellow Rust	MS	Pixel-Based Segmentation Models	U-Net	Recall = 0.926, F-Score = 0.92
[11]	Stem Rust	RGB	Pixel-Based Segmentation Models	DeepLabv3+	F-Score = 0.81
[56]	Yellow Rust	RGB	Pixel-Based Segmentation Models	lr-UNet	Acc = 0.9713
[57]	Yellow Rust	MS	Pixel-Based Segmentation Models	UNet, DF-UNet	Acc = 96.93
[58]	Yellow Rust	HS	Object-Level Classification Models	Inception-ResNet	Acc = 0.85
[24]	Helmintosporium leaf blotch	RGB	Object-Level Classification Models	CNN	Acc = 0.9143

Table 3.4: Summary of deep learning-based methods for wheat disease estimation.

RGB sensors are predominantly paired with deep learning (DL) methods, while MS sensors are more frequently used with machine learning (ML) approaches. This distinction is closely tied to the spatial resolution requirements of DL models, which demand high-resolution images typically captured at lower altitudes using RGB sensors. In contrast, ML models are more tolerant of lower-resolution imagery, often obtained from higher flight altitudes. As illustrated in Figure 10, most DL-based studies used UAV images taken below 20 meters, whereas ML-based studies tended to operate UAVs above 30 meters. The limited use of hyperspectral sensors in both ML and DL methods is likely due to their high cost and the complexity of data processing. These trends underscore the growing preference for cost-effective, high-resolution UAV setups in DL-based crop disease estimation [48].

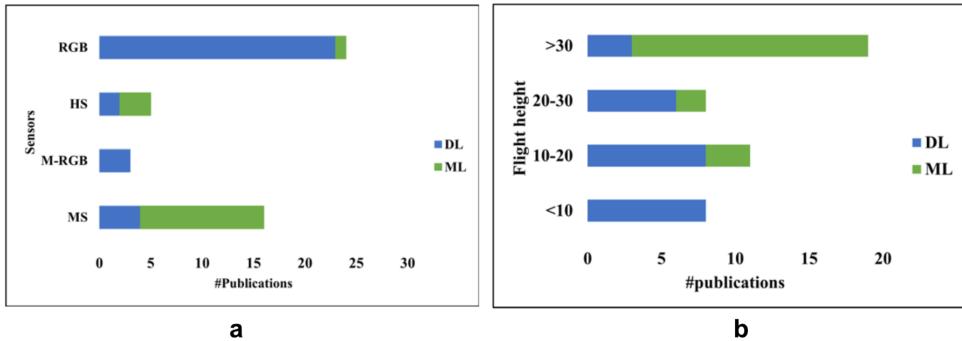


Figure 3.10: The distribution of existing works based on (a) sensors and (b) flight altitudes used in UAV image acquisition. Note that M-RGB denotes modified RGB sensors [48].

3.8 Fusion of Satellite and UAV Data

The fusion of satellite and UAV data presents a promising solution for overcoming the limitations of each platform in agricultural monitoring. While satellites offer broad and frequent coverage, their spatial resolution can be insufficient for detailed field assessments. UAVs provide high-resolution imagery but are limited in coverage and frequency. By combining these complementary strengths, it becomes possible to achieve accurate, timely, and scalable crop monitoring that supports more informed decision-making in precision agriculture [40, 36].

3.8.1 Systematic Categorization of UAV/Satellite Monitoring Methods

UAV/Satellite integration is categorized using a hierarchical decision tree based on three criteria (Figure 3.11). The first distinguishes between weak and strong synergies: Weak synergy involves simple comparisons of UAV and satellite data, while strong synergy combines them to produce more informative outcomes. Among strong synergies, the second criterion differentiates between cases with the same observation target and those with different scales, known as “multiscale explanation,” where UAV data refines or contextualizes broader satellite observations. The third criterion identifies whether models are calibrated using one or both data sources, referred to as "model calibration" (using one to support the other) or "data fusion" (using both to generate new, enhanced data). Data fusion techniques are further divided into pixel-level, feature-level, and decision-level fusion, depending on the depth of integration. Precision agriculture applies all types of synergy strategies, with a slight preference for data fusion. This suggests that precision agriculture benefits from both the spatial detail of UAVs and the broader coverage of satellite data to enhance monitoring and decision-making [3].

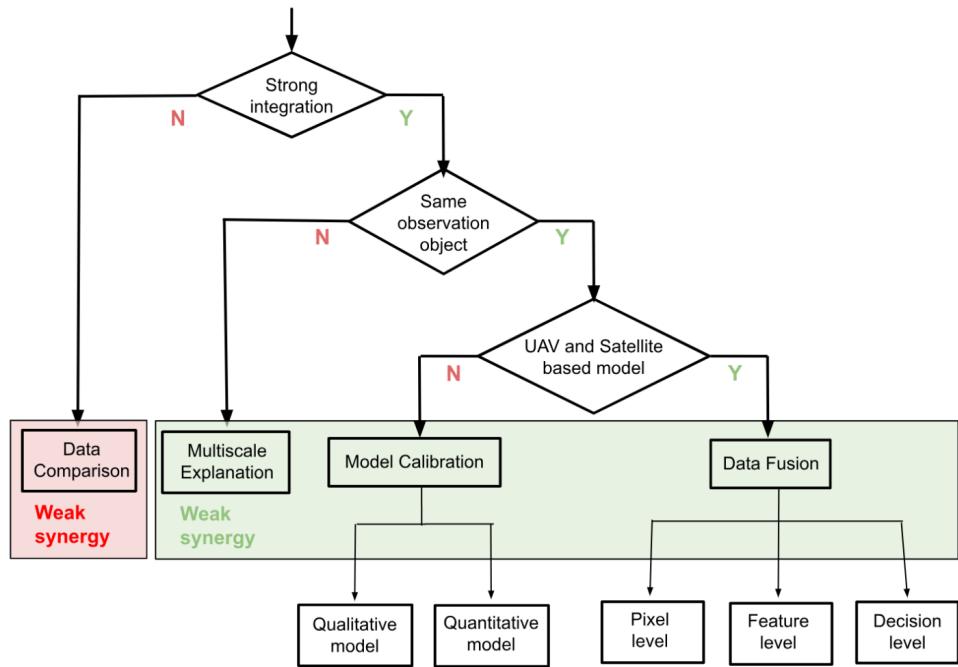


Figure 3.11: Hierarchical decision tree for categorizing UAV/Satellite strategies [3].

Data comparison strategy

The data comparison strategy represents a weak form of UAV/Satellite synergy, where data from both sources are analyzed separately rather than combined (Figure 3.12). These studies highlight the complementary strengths of each platform: satellites offer broad coverage and standardized processing, which is useful for monitoring large or inaccessible areas, while UAVs provide very high spatial resolution (VHSR) at low cost, which is ideal for capturing fine details like crop variability or small water bodies. UAVs are especially

valuable in precision agriculture due to their flexibility in capturing data at critical times for input decisions. Ultimately, the choice between UAV and satellite data depends on the study's scale, goals, and available resources. Notably, 19% of these comparison studies acknowledged the potential for stronger synergies in the future [3].

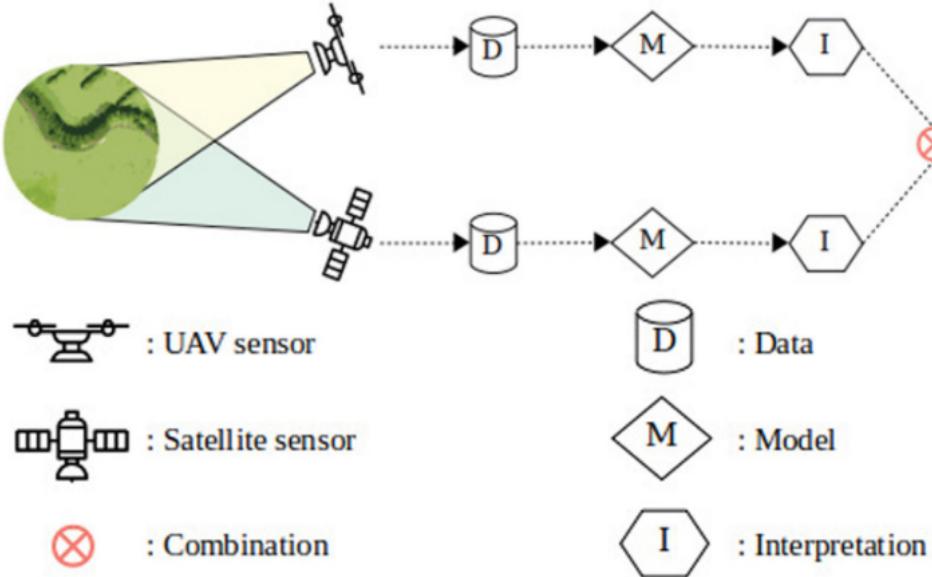


Figure 3.12: Diagram of data comparison strategy [3].

Multiscale explanation strategy

The multiscale explanation strategy leverages the different spatial scales of UAV and satellite data to enhance interpretation (Figure 3.13). Satellites provide a broad, regional context, while UAVs offer detailed, fine-scale information. This strategy enables better understanding by combining wide-area observation with localized, high-resolution insights, often using UAV-derived digital surface models (DSMs) to refine spatial analysis [3].

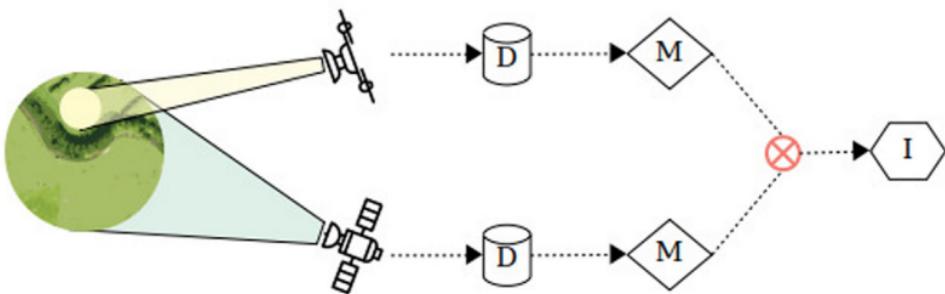


Figure 3.13: Diagram of the Multiscale explanation strategy [3].

Model calibration strategy

The model calibration strategy involves using one data source, often UAV or satellite imagery, to calibrate models built on the other (Figure 3.14). It was the most frequently

used strong synergy approach, especially in ecology and precision agriculture. This strategy can be qualitative, where UAV-derived labels (from expert interpretation, thresholds, or classifications) train satellite-based classification models, or quantitative, where UAV-derived numerical values (e.g., chlorophyll, biomass, height) calibrate regression or unmixing models. In some cases, UAV data replaces traditional ground surveys, serving as the sole reference source. While in-situ data is still used in many studies for validation, UAVs are increasingly relied upon for producing accurate, high-resolution ground truth. The strategy also includes data intercalibration, where one dataset is used to standardize or refine the spectral characteristics of the other [3].

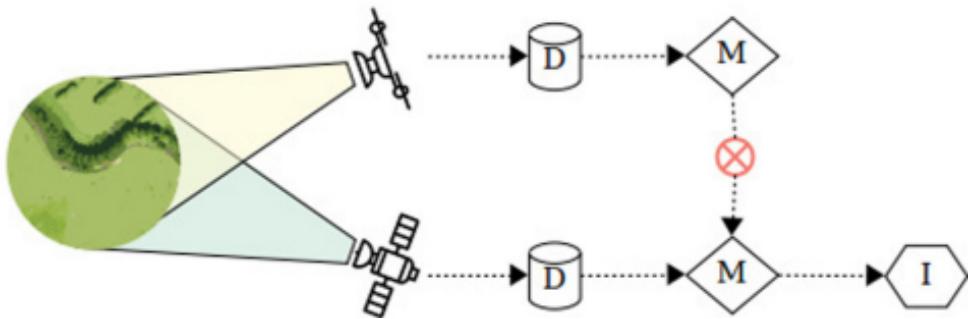


Figure 3.14: Diagram of Model calibration strategy [3].

Data fusion strategy

The data fusion strategy represents the strongest form of UAV/Satellite synergy, aiming to fully integrate both data sources to create enhanced datasets (Figure 3.15). Though less commonly used, it has been explored in precision agriculture for extracting fine-resolution land cover and vegetation traits. Most studies applied pixel-level fusion to combine spatial, spectral, or temporal details, improving classification accuracy, resolution, or time-series completeness. Some studies used feature-level fusion, integrating complementary views from UAVs and satellites for tasks like damage assessment or change detection. While decision-level fusion holds potential, it has not yet been applied in this context. Overall, data fusion enables more accurate and consistent monitoring by leveraging the unique strengths of both platforms [3].

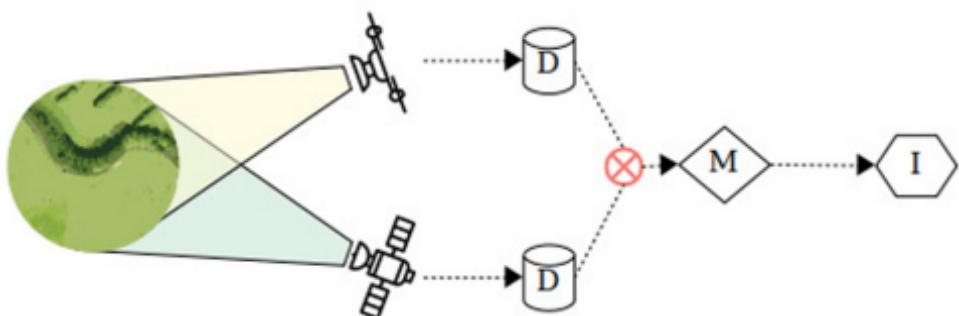


Figure 3.15: Diagram of Data fusion strategy [3].

3.8.2 Challenges of UAV/Satellite Data Fusion

While combining UAV and satellite data offers great potential, it also presents several challenges, including:

- **Limited Exploitation of Synergy:** The potential of UAV/Satellite synergy remains underexploited, as most studies prioritize validation or simple comparisons rather than full integration; advanced strategies like multiscale explanation and data fusion, which preserve and leverage the strengths of both data sources, are still rarely applied [3].
- **Interoperability Issues:** Interoperability is a major challenge due to the lack of standardization in UAV data acquisition and quality assurance, unlike standardized satellite data; variations in sensors, protocols, and conditions further complicate the alignment of multi-source datasets [3].
- **Geometric and Radiometric Misalignment:** Geometric and radiometric misalignment pose a challenge, as misregistration between UAV and satellite data impacts multi-scale analysis; sub-pixel geometric calibration is difficult due to resolution gaps, and radiometric inconsistencies from sensor and environmental differences further hinder seamless integration [3].
- **Technical Complexity:** Fusion of UAV and satellite data remains technically complex, as advanced methods like spatio-temporal and spectral-temporal fusion require specialized expertise, making them less accessible and limiting wider adoption [3].

3.9 Challenges in Integrating Remote Sensing with ML/DL

Despite the promising results obtained from integrating UAV-based remote sensing with deep learning techniques for crop disease detection, several challenges hinder their widespread and effective adoption in real-world agricultural settings. One of the primary limitations is the scarcity of labeled datasets, which are crucial for training deep learning models. Collecting and annotating high-quality images, especially for specific diseases like wheat rusts, is labor-intensive and often inconsistent across studies [48].

Another significant challenge is the high computational complexity associated with training and deploying deep learning models. These models typically require powerful hardware, large memory, and long training times, which can be impractical for use in field conditions or developing regions with limited resources [48].

Furthermore, the selection of optimal models remains non-trivial. The performance of deep learning models can vary significantly depending on the type of crop, disease severity, environmental conditions, and sensor data used. This variation complicates the generalization of models across different agricultural scenarios [48].

3.10 Future Perspectives

UAV-based plant disease detection is an evolving field with considerable potential for advancement through machine learning (ML) and remote sensing technologies. However, current methods, particularly those involving complex deep learning models like CNNs, can be computationally demanding and require significant resources [31]. To address these limitations, future research should focus on developing lightweight, edge-compatible models for real-time, on-device analysis, as well as exploring reinforcement learning and sensor fusion (e.g., combining UAV and satellite data) to enhance disease detection accuracy [48, 31]. Reducing dependence on large annotated datasets through semi-supervised learning and establishing benchmark datasets for standardizing model evaluation are also critical steps forward [48]. Additionally, addressing practical challenges such as the cost of thermal or electrochemical sensors and the technical skill required for drone operation will be essential for widespread adoption, particularly in diverse global agricultural contexts [31].

3.11 Conclusion

The integration of remote sensing with machine learning and deep learning has opened new possibilities for the efficient and accurate detection of wheat diseases. By leveraging data from satellites, UAVs, and other sensing platforms, combined with advanced learning algorithms, it is now possible to monitor crop health at scale, detect early signs of infection, and support timely intervention. This chapter highlighted the main technologies, workflows, and challenges involved in this integration, emphasizing its value in enhancing disease management and promoting sustainable agricultural practices. Continued research and refinement of these methods will be essential to fully realize their potential in real-world farming environments.

General conclusion

The general conclusion goes here.

Bibliography

- [1] Tiago Adão et al. “Hyperspectral imaging: A review on UAV-based sensors, data processing and applications for agriculture and forestry”. In: *Remote Sensing* 9.11 (2017), p. 1110.
- [2] T. Ahmed and N. H. N. Sabab. “Classification and understanding of cloud structures via satellite images with EfficientUNet”. In: *SN Computer Science* 3.1 (2022), p. 99.
- [3] E. Alvarez-Vanhard, T. Corpetti, and T. Houet. “UAV & satellite synergies for optical remote sensing applications: A literature review”. In: *Science of Remote Sensing* 3 (2021), p. 100019. DOI: [10.1016/j.srs.2021.100019](https://doi.org/10.1016/j.srs.2021.100019).
- [4] Laith Alzubaidi et al. “Review of deep learning: concepts, CNN architectures, challenges, applications, future directions”. In: *Journal of Big Data* 8 (2021), pp. 1–74.
- [5] N.R. Aquino et al. “The effect of data augmentation on the performance of convolutional neural networks”. In: *Brazilian Society of Computational Intelligence* 10 (2017).
- [6] Yoshua Bengio, Ian Goodfellow, and Aaron Courville. *Deep Learning*. Vol. 1. Cambridge, MA, USA: MIT Press, 2017, pp. 23–24.
- [7] M. Bhandari et al. “Assessing winter wheat foliage disease severity using aerial imagery acquired from small Unmanned Aerial Vehicle (UAV)”. In: *Computers and Electronics in Agriculture* 176 (2020), p. 105665.
- [8] D. Bohnenkamp, J. Behmann, and A.K. Mahlein. “In-field detection of yellow rust in wheat on the ground canopy and UAV scale”. In: *Remote Sensing* 11.20 (2019), p. 2495. DOI: [10.3390/rs11202495](https://doi.org/10.3390/rs11202495).
- [9] G. Crocioni et al. “Li-ion batteries parameter estimation with tiny neural networks embedded on intelligent IoT micro-controllers”. In: *IEEE Access* 8 (2020), pp. 122135–122146. DOI: [10.1109/ACCESS.2020.3007046](https://doi.org/10.1109/ACCESS.2020.3007046).
- [10] N. Delavarpour et al. “A technical study on UAV characteristics for precision agriculture applications and associated practical challenges”. In: *Remote Sensing* 13.6 (2021), p. 1204.
- [11] J. Deng et al. “Applying convolutional neural networks for detecting wheat stripe rust transmission centers under complex field conditions using RGB-based high spatial resolution images from UAVs”. In: *Comput. Electron. Agric.* 200 (2022), p. 107211. DOI: [10.1016/j.compag.2022.107211](https://doi.org/10.1016/j.compag.2022.107211).

- [12] R. Dhakal et al. “Utilizing spectral, structural and textural features for estimating oat above-ground biomass using UAV-based multispectral data and machine learning”. In: *Sensors* 23.24 (2023), p. 9708.
- [13] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri. “Activation functions in deep learning: A comprehensive survey and benchmark”. In: *Neurocomputing* 503 (2022), pp. 92–108.
- [14] Etienne Duveiller et al. *Wheat Diseases and Pests: A Guide for Field Identification*. CIMMYT, 2012.
- [15] X. Fang, T. Zhen, and Z. Li. “Lightweight Multiscale CNN Model for Wheat Disease Detection”. In: *Applied Sciences* 13.9 (2023), p. 5801.
- [16] U. B. Farook et al. “A Review on Insect Pest Complex of Wheat (*Triticum aestivum* L.)” In: *Journal of Entomology and Zoology Studies* 7.1 (2019), pp. 1292–1298.
- [17] Manuel Figueroa, Kim E. Hammond-Kosack, and Peter S. Solomon. “A Review of Wheat Diseases—A Field Perspective”. In: *Molecular Plant Pathology* 19.6 (2018), pp. 1523–1536.
- [18] D. Garg and M. Alam. “Smart agriculture: A literature review”. In: *Journal of Management Analytics* 10.2 (2023), pp. 359–415.
- [19] S. Ghazal, A. Munir, and W. S. Qureshi. “Computer Vision in Smart Agriculture and Precision Farming: Techniques and Applications”. In: *Artificial Intelligence in Agriculture* (2024).
- [20] A. Guo et al. “Wheat yellow rust detection using UAV-based hyperspectral technology”. In: *Remote Sensing* 13.1 (2021), p. 123.
- [21] W. Haider et al. “A Generic Approach for Wheat Disease Classification and Verification Using Expert Opinion for Knowledge-Based Decisions”. In: *IEEE Access* 9 (2021), pp. 31104–31129. DOI: [10.1109/ACCESS.2021.3059421](https://doi.org/10.1109/ACCESS.2021.3059421).
- [22] R. Heidarian Dehkordi et al. “Monitoring wheat leaf rust and stripe rust in winter wheat using high-resolution UAV-based red-green-blue imagery”. In: *Remote Sensing* 12 (2020), p. 3696. DOI: [10.3390/rs12223696](https://doi.org/10.3390/rs12223696).
- [23] M.I. Hossen et al. “Transfer learning in agriculture: a review”. In: *Artificial Intelligence Review* 58.4 (2025), p. 97.
- [24] H. Huang et al. “Detection of helminthosporium leaf blotch disease based on UAV imagery”. In: *Appl. Sci.* 9 (2019), p. 558. DOI: [10.3390/app9030558](https://doi.org/10.3390/app9030558).
- [25] G. Idoje, T. Dagiuklas, and M. Iqbal. “Survey for Smart Farming Technologies: Challenges and Issues”. In: *Computers & Electrical Engineering* 92 (2021), p. 107104.
- [26] O. Jouini et al. “Wheat Leaf Disease Detection: A Lightweight Approach with Shallow CNN Based Feature Refinement”. In: *AgriEngineering* 6.3 (2024), pp. 2001–2022.
- [27] P. L. Kashyap et al. *Identification Guide for Major Diseases and Insect-Pests of Wheat*. Tech. rep. 18. Technical Bulletin, 2018.
- [28] J. D. Kelleher. *Deep Learning*. MIT Press, 2019.
- [29] Nikhil Ketkar and Eloi Santana. *Deep Learning with Python*. Vol. 1. Springer, 2017.

- [30] I. H. Khan et al. "Early detection of powdery mildew disease and accurate quantification of its severity using hyperspectral images in wheat". In: *Remote Sensing* 13.18 (2021), p. 3612. DOI: [10.3390/rs13183612](https://doi.org/10.3390/rs13183612).
- [31] L. Kouadio et al. "A review on UAV-based applications for plant disease detection and monitoring". In: *Remote Sensing* 15.17 (2023), p. 4273. DOI: [10.3390/rs15174273](https://doi.org/10.3390/rs15174273).
- [32] Andrej Krenker, Janez Bešter, and Andrej Kos. "Introduction to the Artificial Neural Networks". In: *Artificial Neural Networks: Methodological Advances and Biomedical Applications*. InTech, 2011, pp. 1–18.
- [33] A. Li et al. "Water surface object detection using panoramic vision based on improved single-shot multibox detector". In: *EURASIP Journal on Advances in Signal Processing* 2021 (2021), pp. 1–15.
- [34] J. Li et al. "Implicit sparse regularization: The impact of depth and early stopping". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 28298–28309.
- [35] W. Li et al. "Monitoring maize canopy chlorophyll content throughout the growth stages based on UAV MS and RGB feature fusion". In: *Agriculture* 14.8 (2024), p. 1265.
- [36] Y. Li et al. "A spatio-temporal fusion framework of UAV and satellite imagery for winter wheat growth monitoring". In: *Drones* 7.1 (2022), p. 23. DOI: [10.3390/drones7010023](https://doi.org/10.3390/drones7010023).
- [37] L. Liu et al. "Monitoring Wheat Fusarium Head Blight Using Unmanned Aerial Vehicle Hyperspectral Imagery". In: *Remote Sensing* 12.23 (2020), p. 3811. DOI: [10.3390/rs12233811](https://doi.org/10.3390/rs12233811).
- [38] B. Lu et al. "Recent advances of hyperspectral imaging technology and applications in agriculture". In: *Remote Sensing* 12.16 (2020), p. 2659.
- [39] H. Ma et al. "Using UAV-Based Hyperspectral Imagery to Detect Winter Wheat Fusarium Head Blight". In: *Remote Sensing* 13 (2021), p. 3024. DOI: [10.3390/rs13153024](https://doi.org/10.3390/rs13153024).
- [40] M. Maimaitijiang et al. "Crop monitoring using satellite/UAV data fusion and machine learning". In: *Remote Sensing* 12.9 (2020), p. 1357. DOI: [10.3390/rs12091357](https://doi.org/10.3390/rs12091357).
- [41] Y. R. Mehta. *Wheat Diseases and Their Management*. Springer, 2014. DOI: [10.1007/978-3-319-06477-0](https://doi.org/10.1007/978-3-319-06477-0).
- [42] D. Olson and J. Anderson. "Review on unmanned aerial vehicles, remote sensors, imagery processing, and their applications in agriculture". In: *Agronomy Journal* 113.2 (2021), pp. 971–992.
- [43] Q. Pan et al. "A deep-learning-based approach for wheat yellow rust disease recognition from unmanned aerial vehicle images". In: *Sensors* 21.6540 (2021). DOI: [10.3390/s21206540](https://doi.org/10.3390/s21206540).
- [44] S.K. Phang et al. "From satellite to UAV-based remote sensing: A review on precision agriculture". In: *IEEE Access* (2023). [Preprint].
- [45] P. Purwono et al. "Understanding of convolutional neural network (CNN): A review". In: *International Journal of Robotics and Control Systems* 2.4 (2022), pp. 739–748.

- [46] Shaoqing Ren et al. “Faster R-CNN: Towards real-time object detection with region proposal networks”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.6 (2016), pp. 1137–1149.
- [47] Z. Ren and C. Du. “A review of machine learning state-of-charge and state-of-health estimation algorithms for lithium-ion batteries”. In: *Energy Reports* 9 (2023), pp. 2993–3021.
- [48] T.B. Shahi et al. “Recent Advances in Crop Disease Detection Using UAV and Deep Learning Techniques”. In: *Remote Sensing* 15.9 (2023), p. 2450. DOI: [10 . 3390/rs15092450](https://doi.org/10.3390/rs15092450).
- [49] I. Sharma et al. “Enhancing Wheat Production—A Global Perspective”. In: *The Indian Journal of Agricultural Sciences* 85.1 (2015), pp. 03–13.
- [50] J. Singh et al. “Important Wheat Diseases in the US and Their Management in the 21st Century”. In: *Frontiers in Plant Science* 13 (2023), p. 1010191.
- [51] R.P. Sishodia, R.L. Ray, and S.K. Singh. “Applications of remote sensing in precision agriculture: A review”. In: *Remote Sensing* 12.19 (2020), p. 3136.
- [52] J. Su et al. “Aerial visual perception in smart farming: Field study of wheat yellow rust monitoring”. In: *IEEE Trans. Ind. Inform.* 17 (2020), pp. 2242–2249. DOI: [10 . 1109/TII.2020.2967982](https://doi.org/10.1109/TII.2020.2967982).
- [53] C. Tan et al. “A survey on deep transfer learning”. In: *Artificial Neural Networks and Machine Learning – ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4–7, 2018, Proceedings, Part III*. Springer International Publishing, 2018, pp. 270–279.
- [54] Q. Wan et al. “Optimizing UAV-based uncooled thermal cameras in field conditions for precision agriculture”. In: *International Journal of Applied Earth Observation and Geoinformation* 134 (2024), p. 104184.
- [55] D. Zhang et al. “Precision Agriculture: Temporal and Spatial Modeling of Wheat Canopy Spectral Characteristics”. In: *Agriculture* 15.3 (2025), p. 326. DOI: [10 . 3390/agriculture15030326](https://doi.org/10.3390/agriculture15030326).
- [56] T. Zhang et al. “Ir-unet: Irregular segmentation u-shape network for wheat yellow rust detection by UAV multispectral imagery”. In: *Remote Sens.* 13 (2021), p. 3892. DOI: [10 . 3390/rs13193892](https://doi.org/10.3390/rs13193892).
- [57] T. Zhang et al. “Wheat yellow rust severity detection by efficient DF-UNet and UAV multispectral imagery”. In: *IEEE Sens. J.* 22 (2022), pp. 9057–9068. DOI: [10 . 1109/JSEN.2022.3154679](https://doi.org/10.1109/JSEN.2022.3154679).
- [58] X. Zhang et al. “A deep learning-based approach for automated yellow rust disease detection from high-resolution hyperspectral UAV images”. In: *Remote Sens.* 11 (2019), p. 1554. DOI: [10 . 3390/rs11131554](https://doi.org/10.3390/rs11131554).
- [59] Hong Zhao et al. “Research on a learning rate with energy index in deep learning”. In: *Neural Networks* 110 (2019), pp. 225–231.
- [60] Zhi-Qi Zhao et al. “Object detection with deep learning: A review”. In: *IEEE Transactions on Neural Networks and Learning Systems* 30.11 (2019), pp. 3212–3232.
- [61] W. Zhu et al. “Using UAV Multispectral Remote Sensing with Appropriate Spatial Resolution and Machine Learning to Monitor Wheat Scab”. In: *Agriculture* 12.7 (2022), p. 1785. DOI: [10 . 3390/agriculture12071785](https://doi.org/10.3390/agriculture12071785).

Appendix A

Dependencies and libraries

annex a