Computer Vision for Plant Pathology: Integrating Biology with Vision AI

Jamie Robert Sykes

Doctor of Philosophy.

University of York

Computer Science

July 2025

Abstract

Crop diseases threaten to trigger cascading failures across global food systems, devastating ecosystems, economies and livelihoods. In such historical agricultural collapses, smallholder farmers in low-resource settings have been disproportionately harmed. As agricultural technology becomes democratised, we see new opportunities to develop tools that promote stability and incremental improvement. While numerous Artificial intelligence (AI) and Machine learning (ML) models have been developed for plant disease detection, most rely on generic architectures trained on limited or synthetic datasets. As such, they struggle to detect early or non-visible symptoms, and don't run efficiently on low-power hardware. This thesis addresses early disease detection in cocoa crops by integrating advanced computer vision (CV) techniques with biological knowledge, offering effective, efficient, and transparent AI models tailored to resource-constrained environments. Key contributions include PhytNet, a custom lightweight convolutional neural network (CNN) architecture designed for accurate diagnosis with limited hardware and training data. We investigate spectroscopy and infrared (IR) imaging as sources of complementary signal beyond the visible spectrum, finding infrared imagery can be competitive in image-based classification. We leverage semi-supervised learning and a novel dynamic focal loss (DFLoss) to direct model attention toward difficult-to-detect symptoms, and enhance interpretability through gradient-weighted class activation mapping (Grad-CAM) visualisations, enabling validation of model focus against biological symptoms. We also present a new high-quality benchmark dataset of 7,220 images of diseased and healthy cocoa trees, offering a greater and more realistic challenge than existing benchmarks like PlantVillage. Extensive experiments demonstrate that tailored architecture design and advanced training procedures improve detection accuracy, generalisation and resource efficiency, while non-visible sensing analyses narrow where complementary signal is most plausibly useful. By bridging computer science, plant pathology, and exploratory non-visible sensing, this work yields new methods for AI-driven disease surveillance tools that can help smallholder farmers protect crops and promote sustainable agriculture, highlighting the societal and ecological relevance of such interdisciplinary work.

Contents

List of Figures

List of Tables

To my family and friends, with thanks for their patience and support.

Acknowledgements

This work would not have been possible without the generous support of many individuals and institutions. I am deeply grateful to Prof. Daniel W. Franks and Prof. Katherine J. Denby for their outstanding guidance, insightful feedback, and unwavering encouragement throughout every stage of this research.

I would like to thank the Doctoral Centre for Safe, Ethical and Secure Computing at the University of York for providing the studentship that funded this work, under the supervision of Professor Leandro Soares Indrusiak. Additional financial support was provided by the University of York-Engineering and Physical Sciences Research Council (EPSRC) Impact Accelerator Fund, which enabled crucial fieldwork and data-collection activities.

My thanks go to Barry Callebaut, especially Maylin Yoong and Martin Gilmour, for facilitating access to commercial cocoa plantations, and to the team at Instituto Nacional de Investigaciones Agropecuarias (INIAP) Ecuador, especially Dr. Rey Gastón Loor Solórzano and Dr. Danilo Vera Coello, for their invaluable assistance in coordinating and conducting on-site data collection. I am also grateful to the staff at the International Cocoa Quarantine Centre (ICQC) (Reading, United Kingdom (UK)) for providing expertise and six cocoa trees.

I also wish to acknowledge the University of York Computer Science Department for providing a fantastic, supportive and stimulating environment and home for this research. Special thanks to colleagues in the Vision Graphics and Learning Research group, whose technical discussions and collaborative spirit greatly enriched this work.

Declaration

I declare that this thesis is a presentation of original work and I am the sole author. This work has not previously been presented for an award at this, or any other, University. All sources are acknowledged as References.

Parts of the research described in this thesis have previously published in:

- Sykes, J. R., Denby, K. J., & Franks, D. W. (2024). Computer vision for plant pathology: A review with examples from cocoa agriculture. *Applications in Plant Sciences, 12*(2), e11559. `https://doi.org/10.1002/aps3.11559`
- Sykes, J. R., Denby, K. J., & Franks, D. W. (2025). Tailoring convolutional neural networks for custom botanical data. *Applications in Plant Sciences*, e11620. `https://doi.org/10.1002/aps3.11620`
- Sykes, J. R., Denby, K. J., & Franks, D. W. (2025). Improving computer vision for plant pathology through advanced training techniques. *Applications in Plant Sciences, 13*(3), e70010. `https://doi.org/10.1002/aps3.70010`

Chapter 1

Introduction

> "March away from the sound of the guns. Observe from a distance, but do not join the fray. Make a fray of your own. Once you have settled on a specialty, and the profession you can love, and you've secured opportunity, your potential to succeed will be greatly enhanced if you study it enough to become an expert."
>
> *E. O. Wilson*, Letters to a Young Scientist

1.1 Motivation

Modern computer vision (CV), typically powered by machine learning (ML) and artificial intelligence (AI), is now used for a variety of tasks in agriculture, botany, and ecology. This application of CV will be the central focus of this thesis. But before we dive in, let's define these three key terms and make clear how they relate to each other: Computer vision is the field of study that enables machines to interpret and extract meaningful information from images and video. Artificial intelligence (AI) refers to computational systems that perform

tasks normally requiring human intelligence, such as perception, reasoning, and decision-making. Machine learning is a subset of AI in which models learn patterns from data to make predictions or decisions without explicit rule-based programming. Tasks now delegated to CV systems include plant health assessments [1], identification of weeds [2], identification of drought-prone areas of land [3], yield prediction [4], and detection of defects or bruising in fruits and vegetables [5]. We are seeing substantial improvement in the efficiency of CV techniques [6, 7, 8] and, at least for now, computational resources continue to become more affordable [9]. As a result, CV is becoming available to whole industries, not just areas of the highest commercial value. For example, AI has been used with increasing regularity for tasks specific to cocoa (*Theobroma cacao L.*), such as the exploration and optimisation of aroma profiles [10], monitoring of cocoa bean fermentation [11, 12], and bean quality classification [13]. Large research and development budgets for areas such as wheat (*Triticum aestivum L.*) production have allowed for the use of unpiloted aerial vehicle photography to identify disease outbreaks [14, 15] and the use of multispectral satellite photography to monitor outbreaks of yellow rust (*Puccinia striiformis*) from space [16]. Yet the application of AI to sectors with fewer financial resources has had to take a different form. Onboard graphics processing units (GPUs) can run large neural networks locally, analysing image data from farm machinery in real-time, while fast internet connections can be used to run the same large models remotely [17]. By contrast, the implementation of AI in poorer sectors must rely on older hardware, edge devices, and older-model smartphones. This means that an emphasis must be placed on the ultra-low-cost implementation and high computational efficiency of algorithms. This provides us with the opportunity and motivation to steer the AI field away from brute-force computing and toward more nuanced and efficient approaches.

The cultivation of cocoa represents a prime example of a sector that could benefit greatly from non-intrusive and highly optimised CV disease detection. As such cocoa will be used as an example application domain throughout this thesis, though the tools and techniques we discuss are applicable to any crop. The International Cocoa Organisation estimates that up to 38% of the global cocoa crop is lost to disease annually, with over 1.4 million tonnes of cocoa lost to just three diseases in 2016 [18, 19]. Additionally, international disease spread has been devastating to this industry in the past and could be again in the future [20, 21]. Following the loss of a cocoa crop to witches' broom disease (WBD) (*Moniliophthora perniciosa*), a plot of land will typically be cleared of forest, and what may have been an

otherwise robust agroforestry system will be replaced with a monoculture [22, 21]. This disease is therefore not only capable of devastating the livelihoods of whole communities of cocoa farmers, eliminating 50–90% of their crop [21], but it is also destructive to local biodiversity and has a significant negative impact on the carbon capture potential of the land [23]. Such loss of Amazonian Forest is a driver of climate change, causing positive feedback and exacerbating this global crisis [24].

A review from 1986 on the use of systemic fungicides to tackle oomycetes, such as *P. spp.*, highlights concern about damage to the environment and human health by pesticides such as methyl bromide, which are still in use today [25] (*n.b.* Oomycetes are fungus-like water moulds that cause many destructive plant diseases but are biologically distinct from true fungi). These concerns and those of the pesticide resistance [26] are still present 37 years later. The use of CV and AI for targeted application and calibration of pesticide dose is beginning to have massive beneficial effects in this area across the agriculture industry.

It is estimated that from 2016 to 2026, the number of smartphone users will have doubled from approximately 3.7 billion people to 7.5 billion [27]. Therefore, the minimum necessary hardware to run CV models is largely in place. Now we need only develop, validate and deploy the CV models to have great potential for impact with little monetary input. In this thesis, we explore how best to achieve this.

1.2 The Economic and Social Importance of Cocoa

The importance of the cocoa industry to the economies of Ghana and the Ivory Coast cannot be overstated. As of 2012, cocoa exports represented 25% of Ghana's income [28] and in 2008 cocoa accounted for 35% of the total exports of the Ivory Coast [29]. In the growing year 2021-2022, Ghana produced 1.047 million tonnes of cocoa and Ivory Coast, as the largest global producer of cocoa, produced 2.248 million tonnes. In the same year, the third-largest producer of cocoa globally, Ecuador, produced only 365,000 tonnes, making it a distant third and comparable to other countries such as Cameroon and Nigeria [30]. More recent reporting suggests that this gap may close rapidly: Reuters reported in September 2025 that Ecuador was on course to produce more than 650,000 metric tons in the 2026/27 season and

could surpass Ghana as the world's second-largest cocoa grower [31]. Until 1921, when the introduction of WBD decimated the Ecuadorian cocoa trade, Ecuador was the largest global producer of cocoa. Yet, despite the low productivity of Ecuadorian cocoa trees [29], and the presence of WBD and frosty pod rot (FPR) (*Moniliophthora roreri*) [32], Ecuador remains the largest global producer of high-quality cocoa [33]. For a brief period in the 1980s, prior to the introduction of WBD, cocoa was the second-largest export from Brazil, after coffee [34]. The collapse of the Brazilian cocoa trade due to WBD led to socioeconomic upheaval, abandonment of farms, clear-cutting of plantations, increased unemployment and migration of workers to the cities [35]. If nothing is done to prevent further spread of these diseases, the events that unfolded in Ecuador and Brazil are likely to occur again in Ghana and the Ivory Coast. Alternatively, with improved disease control technologies, the persistent Ecuadorian and Brazilian cocoa industries could grow back towards their former status as major producers, and we could safeguard the Ghanaian and Ivory Coast crops. Herein lies an opportunity, do we work to support and grow this $200 billion+ global industry that employs more than 5 million people [36] or do we let it collapse? Many cocoa farmers have an alternative industry to which these resilient people may turn when cocoa production is not economically viable. That industry is the illegal, yet highly lucrative, cultivation of coca (*E. coca*) [37] which grows in identical conditions to cocoa. Through research like that presented here, we can provide these farmers with the tools they need to bring prosperity to their communities without becoming involved in the production of cocaine, an activity that puts people at great risk of violence from criminal organisations, national law enforcement and right-wing paramilitary groups [38]. A recent example of such danger was the March 2026 bombing of a dairy farm in San Martín, Ecuador, after it was mistakenly described as a drug camp [39].

1.3 Ethical Considerations

The ethical implications of applying CV and AI technologies to agriculture are multifaceted and warrant careful consideration. Technologies such as CV and AI are inherently subject-agnostic, *i.e.* they encode no intrinsic distinction between legal and illegal activities, nor between morally desirable and undesirable outcomes [40]. This neutrality creates ethical ten-

sion when such technologies can be applied to biologically similar systems that exist within very different legal, social, and political contexts. In this thesis, computer vision methods are developed to support cocoa cultivation through improved disease detection and crop monitoring. The ethical justification for this work is grounded in its intended application: supporting the resilience and economic viability of a global cocoa industry and sustaining the livelihoods of more than five million smallholder farmers worldwide [36][41]. However, alternative crops such as coca (*Erythroxylum coca*) support a highly lucrative but illegal cocaine economy, and the methods developed here could just as easily be applied to coca as cocoa [37]. When considering the ethics of high-tech cultivation of coca, it is essential to distinguish the coca plant from refined cocaine products. The coca plant has been cultivated and consumed for millennia by Indigenous Andean communities for cultural and medicinal purposes, including the mitigation of altitude sickness [42]. When used in its naturally occurring leaf form, coca does not produce the intense psychoactive or addictive effects associated with cocaine hydrochloride or crack cocaine, which arise from chemical extraction and concentration processes [43]. More broadly, plant-derived psychoactive substances, for example from coca, coffee (*Coffea* spp.), cannabis (*Cannabis* spp.), psilocybin-producing fungi (e.g. *Psilocybe* spp.) or ayahuasca (commonly prepared from *Banisteriopsis caapi* in combination with *Psychotria viridis*), exist on a continuum of pharmacological risk and social harm. While cocaine undoubtedly poses greater potential for harm, this continuum extends much further to accommodate highly dangerous substances such as fentanyl or methamphetamine [44]. The key point here is that legal classifications and public perceptions of these substances are strongly shaped by domestic and international political frameworks rather than by pharmacology or population-level harm alone [45]. The authors of this work therefore reject a blanket ethical prohibition on the application of CV or AI technologies to entire categories of organisms based solely on their circumstantial association with illegal drug production. Such an embargo would erase substantial cultural, medical, and historical contexts and conflate biological organisms with downstream criminal economies. The scope and intent of this research are explicitly aligned with supporting agricultural systems, improving farmer livelihoods, and reducing incentives for engagement in high-risk economies [46].

1.4 The Ecological Importance of cocoa

Cocoa is grown under a continuum of systems, ranging from monocultures (fig. 1.1 a) to truly diverse indigenous agroforestry systems wherein cocoa trees are interspersed with old-growth forests (fig. 1.2). The most common systems are full monocultures and those where shade trees are cultivated and harvested alongside cocoa [22] (fig. 1.1 a & b). (*n.b.* Agroforestry refers to the intentional integration of trees with crops or livestock to provide ecological and economic benefits.) Although biodiversity tends to be lower in cocoa plantations than in primary forest, biodiversity in cocoa plantations is still typically higher than in any other agricultural land use system [47]. In countries where high-tech agriculture is common, more complex systems like agroforestry and mixed cropping are becoming increasingly popular under the banner of regenerative agriculture [48]. This is because, in Western agriculture, we are beginning to recognise the potential of these diverse systems to increase yields, promote economic stability, conserve soil structure and inhibit pest spread and resistance [49, 48, 50]. Likewise, the potential benefits of such systems to cocoa agriculture are undeniable. The role of shade trees in cocoa agriculture is to moderate light, air temperature, humidity and wind, to hamper pest dynamics, to produce products such as timber and rubber, and to maintain soil health [51]. A well-designed configuration of shade trees, with high species diversity and nitrogen-fixing legumes, has been shown to be beneficial in managing FPR [52]. In coffee agroforestry, multiple studies have shown the advantages of increased plant biodiversity, shade trees, soil conservation and reduced use of pesticides in these perennial crops [53, 54]. However, despite the abundant benefits and a trend towards these more traditional methods in European and North American agriculture, cocoa farmers are under pressure to intensify production to maximise short-term gains. This pressure comes from increasing demand for cocoa and growing human populations in cocoa-producing regions around the world. The result of this is that long-term sustainable benefits are being sacrificed [33, 47].

The pace of technological and agricultural advancement in cocoa-producing regions has been slow. As a result, these healthy and robust semi-indigenous farming systems and practices still exist in this industry and do not require reinventing as they do in other agricultural sectors, *i.e.* we have the opportunity to promote these practices now before they are lost. Non-intrusive technologies such as ML and CV are ideally suited to provide informed solu-

tions in cases such as pest management and can be seamlessly integrated into such indigenous farming systems. It is a matter of fact that we must increase cocoa production and, in the short term, monocultures have historically allowed us to increase crop production to great effect. However, with the effects of rapid climate change looming, we can no longer afford to be so short-sighted. Crops such as cocoa represent a system of agriculture that can be developed with great beneficial effect for local biodiversity and the livelihoods of the communities that depend on it, but only if well managed.

(a) (b) (c)

Figure 1.1: (a) cocoa monoculture, (b) simple cocoa-coconut agroforestry and (c) simple cocoa-rubber agroforestry. [55]

Figure 1.2: Indigenous cocoa agroforestry. Created by thinning primary-growth forest and subsequently cultivating cocoa trees. [56]

1.5 Global Distribution of Cocoa Disease

The global distribution of cocoa disease and its potential for spread is a complex matter. It is intertwined with the influence that this crop has had on human society and the history of global trade. Arguably, the most economically important disease in cocoa is black pod rot (BPR), which is caused by *Phytophthora megakarya* or *Phytophthora palmivora* [57, 58, 59]. *Phytophthora*, which translates to "plant destroyer" in English, is a genus which is primarily composed of 17 pathogenic species that cause serious loss in many crop species [60]. The most intensely researched of these species are *P. infestans, P. cinnamomi, P. megakarya* and *P. palmivora* [61]. *P. megakarya* is currently confined to Africa but species distribution models show that, with increasing global trade, further international spread is likely [60]. This is highly concerning to the cocoa industry as *P. megakarya* has been shown to be three times more destructive than *P. palmivora*, which is found in all cocoa-producing regions [62].

FPR is currently found throughout Central America and in northern South America, west of the Andes [63]. As well as the threat of *P. megakarya* spreading to America or Asia, the potential international spread of FPR gives cause for concern. This is because, despite its many characteristics which make it a proficient coloniser [63], it has not yet spread to Brazil or Africa [57].

M. perniciosa, which causes WBD in cocoa, likely evolved alongside *T. cacao* in the Amazon basin [63]. Like FPR, WBD is present in several of the top ten cocoa-producing countries but is notably absent from the top two, Ghana and the Ivory Coast [30]. WBD appeared in Brazil in the 1970s and was spread deliberately to Brazil's main cocoa-producing region, Bahia, in 1989 [35]. This act, which is considered to have been politically motivated [35], caused Brazil to go from the second-largest cocoa producer, behind the Ivory Coast, to its current position in sixth place [34, 30]. In 2001 WBD appeared in Espírito Santo, Brazil and spread rapidly across the state [64]. This rapid spread and the resultant economic devastation are a testament to the need for capable, autonomous early identification and tracking of these diseases across borders.

1.6 Progression of Disease Symptoms

With both cultural and chemical controls, timing is key. Fungicides have much greater efficacy when applied early in disease development [65] and many have no curative activity at all. Likewise, if diseased plant tissue is removed from a crop as soon as possible, the loss of a whole season's harvest can be avoided.

In FPR, *Moniliophthora roreri* spores germinate readily on the surfaces of damp cocoa pods, though they can also germinate in the absence of water [66]. The conidia, asexual spores that enable rapid dispersal and infection, produced by *M. roreri* are the only known means of propagation and infection in this pathogen. The conidia do not last long in soil but, when on a tree, they can remain viable for up to 9 months, *i.e.* long enough to last between crops and seasonal dry periods [67]. Raindrops or very slight wind are enough to dislodge conidia from hanging pods. Intermittent winds and convection can then carry spores more than 1 km [68]. Insects are not considered important in transporting spores or allowing them access through the outer layers of pods [67]. Once germinated, spores proceed to infect the pod in a process that takes several hours. Once infection is complete, young pods will begin to show visible symptoms within 4-7 weeks [69], but then those symptoms can disappear until late in pod development. FPR primarily causes symptoms on the cocoa pod but can also cause seed rot. Symptoms of this disease include abnormal pod shape, discolouration, extensive mould, black or brown lesions and mummification [70].

Unlike the two fungal diseases discussed here, BPR is caused by one of two oomycetes: *P. megakarya* and *P. palmivora*. These two pathogens cause similar symptoms to each other, except that the former causes lesions with irregular borders and the latter causes typically smaller lesions with smooth borders [71]. Following initial infection by these pathogens, the whole pod will typically be covered by lesions in 3-5 days [71] and beans will become infected after 15 days. All commercial value will then have been lost. Symptoms of BPR are not restricted to the pod and can cause damage to the roots, seeds and stem. Symptoms include brown or black lesions, mummification, premature dropping of fruit, soft rot of the root cortex, seed rot, stem canker, damping off and whole plant death [71]. The spores of *P. palmivora* and *P. megakarya* are not readily released by air movement but by rainfall. Many *P. spp.* are soil-borne [72], though some, such as *P. infestans*, are more tolerant to

desiccation and will release spores readily in the presence of wind [73]. Both *P. palmivora* and *P. megakarya* can be dispersed by wind and animal vectors but are primarily splash dispersed [74]. It is said that even if relative humidity is 100%, *P. palmivora* spores will lose viability within minutes [73]. There is also an inverse relationship between the duration of *P. palmivora* spore viability and temperature [73]. Dispersal of *P. palmivora* is typically less than 70 m [75] and dispersal of *P. megakarya* is typically 2.7-6.9 m [76]. This low dispersal distance and the typical progression of disease symptoms being from the ground up is consistent with the characterisation of these two pathogens as being soil-borne and splash-dispersed [58]. These pathogens use invertebrates as their primary means of long-distance dispersal and invertebrate-caused damage and other wounds as a route of infection [77].

The dispersing basidiospores of *M. perniciosa*, the fungus that causes WBD, are sensitive to drying (*n.b.* basidiospores are sexual spores produced on structures called basidia and are a key stage in fungal spread.) However, during the night when humidity is high, they are said to be able to survive wind dispersal of 50-70 km [78]. Unwitting human transportation of infected plant material is likely to be one of the main culprits of long-distance dispersal of this pathogen [78]. Following the initial infection of meristematic tissue such as flower cushions, stem tips or pods, *M. perniciosa* displays a biotrophic growth phase, feeding on living host tissue and suppressing host defences to maintain viability. After 1-2 months of development, necrosis of the plant tissue occurs distal to the point of infection [21]. The fungus then switches to a saprophytic growth phase, feeding on dead organic tissue. During this phase it produces fruiting bodies known as basidiocarps. These fruiting bodies can be continually produced for multiple years on dead plant tissue [69], illustrating the need for phytosanitation. The dried, necrotic tissue caused by this progression of symptoms forms the dry broom for which this disease is named. Like BPR, WBD can infect all parts of the plant. Symptoms of WBD can include oozing, scabbing, pitting, malformed skin and abnormal shape in the fruits, as well as black or brown lesions on the fruit. It can also cause malformation and lesions on the leaves, seeds and inflorescence, stem cankers, distortion in stem growth and discolouration of the bark [79].

To consolidate the above discussion, fig. 1.3 presents one early-stage and one late-stage example for each of the three focal diseases, and table 1.1 summarises field-relevant lifecycle

progression and symptom regions. Symptoms of diseases such as these are incredibly variable and can be influenced by a huge variety of factors, including the age of the plant, the age of the infected tissue, the cultivar, the environment, and the presence of other pathogens [70] [71] [79]. This variability, which cannot reasonably be portrayed in images here, is a major challenge for disease detection and management. However, it also means that there are many potential avenues for early detection.

Figure 1.3: Representative early (below) and late (above) symptom examples for black pod rot, frosty pod rot and witches' broom, with early diseased regions circled.

With each of these diseases, there is considerable lead time between the initial infection and the appearance of human visible symptoms. It is this window of opportunity that this work aims to utilise to inform the targeted use of pesticides and phytosanitation. This informed action could save commercially valuable seeds and potentially preserve whole trees or plantations. In the weeks between infection and symptom development, farmers can be blissfully unaware of the damage occurring to what may be their sole source of income. However, with automated disease detection systems in place, entire communities of farmers could be saved from devastating losses.

Table 1.1: Summary of lifecycle progression and visible symptom regions for the three major cocoa diseases discussed in this section.

Disease	Lifecycle progression	Early symptoms	Late symptoms
Frosty pod rot	Conidia disperse to pod surfaces, germinate, penetrate pod tissue, then proceed through a latent phase before intense external sporulation and tissue collapse.	Local pod discolouration and swelling/deformation near the infection site.	Extensive white sporulating surface growth, broad necrosis and pod mummification.
Black pod rot	Splash-dispersed inoculum infects pods, lesions expand rapidly over days, then rot progresses into beans and can continue across multiple plant tissues.	Small dark, water-soaked lesions on pod skin, often beginning in lower-canopy fruit.	Whole-pod black-/brown rot, deep tissue breakdown and loss of bean quality.
Witches' broom	Basidiospores infect meristematic tissue, followed by a biotrophic phase and then a necrotic saprophytic phase with recurrent basidiocarp production.	Abnormal hypertrophic growth in shoots, cushions or pods at localised meristematic regions.	Dry broom structures, necrotic stems/cushions and severe organ deformation.

1.7 Current Methods of Control

The removal of diseased plant tissue, also known as phytosanitation, has been shown to reduce pod loss and delay the onset of witches broom disease [80]. Weekly removal of diseased pods has also been shown to decrease the instances of WBD, BPR and FPR by 14-66%, depending on the disease [81]. However, when using this method to control witches' broom disease, as many as 95% of brooms may have to be removed to achieve a 50% reduction in pod loss [80]. Additionally, many farmers lack the equipment to automate this process, which means it remains a time-consuming and expensive task [35]. The result is that phytosanitation is often neglected [20] and diseases are allowed to proliferate before action is taken.

Fungicides can be effective against both the fungal and oomycete pathogens of cocoa. Copper-based fungicides have been shown to be effective against FPR and BPR [82] and systemic fungicides such as metalaxyl are widely used [82]. However, metalaxyl has had serious prob-

lems with resistance when used to treat *P. infestans* in potatoes [83], which exemplifies a prominent risk for many pesticides. Another effective fungicide used on cocoa is systhane. The International Cocoa Quarantine Centre (ICQC) operations manual mandates that systhane must be applied to any plant material that has been newly imported and is showing signs of fungal disease. These agrochemicals can be effective at high doses and are essential to the production of cocoa [84]. However, as most cocoa is produced by smallholders with just a few hectares in production [85], these fungicides represent a cost that most struggle to afford. This is evidenced by the fact that only 9-21% of Ghanaian cocoa farmers apply fungicides [86]. Additionally, as of 2008, European Union regulations limit the maximum fungicide residue levels allowed on cocoa beans. This, alongside legislation from the United States of America (USA) and Japan, has greatly restricted the use of pesticides in cocoa production [84, 87].

1.8 Research Aims

The above introduction highlights a multitude of difficulties one may encounter in the management of disease in cocoa. What's more, many, if not all, of these issues apply to crops across arable agriculture, horticulture, and silviculture. It is clear that crop disease control is an immense challenge for modern societies that are committed to the large-scale monoculture model of food production, in which these pathogens thrive.

CV presents a uniquely powerful tool for addressing this challenge. Unlike traditional scouting or laboratory-based diagnostics, CV allows for rapid, non-invasive, and scalable assessment of plant health. It can operate in-field and in real-time, enabling earlier intervention and more targeted management. When implemented on edge devices or low-cost hardware, it also becomes accessible to smallholder farmers who lack access to high-tech infrastructure. This makes CV one of the few technologies that can bridge the gap between cutting-edge science and resource-constrained agricultural systems.

AI and CV may well make this task more manageable, but the application of these tools themselves compounds the complexity of this issue. In this thesis, we aim to disentangle this latter group of complexities so that we might apply CV to the task of crop disease

management with favourable and sustainable results. We begin with a detailed review of the literature surrounding CV theory and its application in plant pathology. We then present an initial chapter establishing baseline methods and evaluation practices for the focal cocoa task, followed by a technical chapter that explores spectroscopy as a complementary sensing modality. Finally, we present two chapters focused on methodological advances for image-based diagnosis: (1) a holistic approach to model development and data acquisition, centred on tailoring a novel convolutional network for a small but highly informative dataset, and (2) the collection of a larger, higher-quality cocoa disease dataset and the application of advanced training procedures that promote improved generalisation and more explicable model behaviour.

1.9 Scope of This Research

This thesis focuses on the development and evaluation of practical computer vision methods for cocoa disease detection in low-resource settings. The core application domain is cocoa pathology, with emphasis on three economically important diseases: BPR, FPR, and WBD.

Methodologically, the scope includes model architecture design and comparison, data acquisition and curation for realistic field conditions, training strategies that improve generalisation and interpretability, and runtime profiling relevant to edge and low-cost deployment constraints. Experimental work is centred on convolutional neural network pipelines, including PhytNet and ResNet18 variants, evaluated using performance metrics, cross-validation, and attention-map-based interpretation.

The following areas are intentionally outside the present scope:

- full commercial product engineering and systems integration;
- multi-season, multi-country deployment trials and longitudinal user-adoption studies;
- controlled agronomic intervention studies that quantify yield change after model-guided spraying;
- molecular or causal validation of every learned visual feature;

- formal regulatory, legal, or macroeconomic impact analysis.

Transformer-first model families and full semantic-segmentation deployment pipelines are reviewed in Chapter 2 but are not the primary experimental focus of this thesis. These boundaries are deliberate and keep the work focused on technically rigorous, biologically informed, and computationally efficient foundations that can be validated and extended in future translational studies.

1.10 Research Questions and Hypotheses

This thesis is organised around one overarching research question and hypothesis, followed by focused sub-questions and sub-hypotheses that are tested across the technical chapters.

- **RQ0 (overarching):** Can biologically informed and computationally efficient computer vision systems provide accurate, generalisable, and interpretable detection of major cocoa diseases under realistic field conditions?
- **H0 (overarching):** A data-centric pipeline that combines tailored model design, biologically informed data acquisition, and training procedures aimed at robust generalisation will outperform generic workflows for real-world cocoa disease detection.

- **RQ1/H1:** Can a tailored lightweight convolutional architecture match or exceed the practical performance of larger off-the-shelf models while reducing computational cost?
- **RQ2/H2:** Do non-visible signals, especially infrared and spectroscopy-derived signatures, improve disease discrimination and early symptom detection compared with standard visible-spectrum imaging alone?
- **RQ3/H3:** Do advanced training procedures, including semi-supervised learning, dynamic focal loss, and a non-cocoa class, improve model fit to difficult real-world cases and reduce overfitting?
- **RQ4/H4:** Can model interpretability and deployment suitability be improved concurrently, such that stronger attention to biologically relevant features is achieved without compromising runtime feasibility for low-resource use?

1.11 Contributions

The principal contributions that emerged from this research are:

1. A tailored lightweight convolutional neural network architecture (PhytNet) and optimisation workflow for plant disease detection under limited-data and low-compute constraints.
2. Biologically informed sensing beyond the visible spectrum, showing that infrared imagery can remain competitive for cocoa disease detection while the field-spectroscopy evidence presented here was inconclusive.
3. The development and curation of a challenging field-relevant cocoa disease image dataset, and the use of this dataset to benchmark practical model behaviour under realistic variability in disease presentation.
4. The development of a novel dynamic focal loss (dynamic focal loss (DFLoss)) as part of this thesis, designed to prioritise difficult examples in multi-class plant disease classification and improve practical generalisation.
5. A training framework that combines semi-supervised relabelling and non-cocoa negatives to improve hard-case classification and support better generalisation in real-world conditions.
6. A multi-criteria evaluation approach that integrates summary metrics, cross-validation, attention-map inspection, and runtime profiling to support defensible model selection for deployment.
7. A conclusive analysis of several popular CV architectures, including ResNet18, EfficientNetB0, MobileNetV3Small and PhytNet, with a truly independent test set. This highlighted overfitting behaviour and high risk of poor generalisation in all of these models.

Taken together, these contributions provide a coherent pipeline for building practical, explainable, and computationally efficient computer vision systems for cocoa disease monitor-

ing, with potential transfer to related crop-pathology settings and many potential pitfalls signposted along the way.

1.12 Thesis structure

The remainder of this thesis is organised as follows.

- **Chapter 2:** reviews the biological, computational, and methodological background needed for the thesis, including plant pathology, computer vision, model evaluation, and deployment considerations.
- **Chapter 3:** reports the preliminary investigations that shaped the later contribution chapters, including exploratory studies in segmentation, outlier filtering, normalisation, and optimisation choices.
- **Chapter 4:** examines MultispeQ measurements and pod reflectance spectra to test whether non-visible signals provide robust discriminatory value for cocoa disease detection and to motivate the later infrared imaging experiments.
- **Chapter 5:** presents the development and evaluation of PhytNet, focusing on architecture design, optimisation, comparative analysis against established models, and evaluation under both visible and infrared image settings.
- **Chapter 6:** introduces advanced training procedures, a larger and more challenging cocoa dataset, and a truly independent test set, evaluating their effects on model fit, generalisation, interpretability, and runtime behaviour.
- **Chapter 7:** draws together the main future research directions arising from the thesis, including deployment, multimodal sensing, continual learning, and broader validation across crops and environments.
- **Chapter 8:** revisits the research questions and hypotheses, synthesises the principal findings of the thesis, and closes with the main limitations, assumptions, and final remarks.

Chapter 2

Background

> "The mouse is a sober citizen who knows that grass grows in order that mice may store it as underground haystacks, and that snow falls in order that mice may build subways from stack to stack."
>
> *Aldo Leopold, A Sand County Almanac*

This review is composed of three main sections. Section 1, "Methods in computer vision," begins with a brief historical and conceptual overview before critically reviewing a wide variety of relevant techniques in machine learning (ML), artificial intelligence (AI), and computer vision (CV) model development and testing. Section 2, "Data acquisition and model testing," discusses techniques for data gathering, data labelling, validation, and biological interpretation. While section 1 focuses on ML/AI theory and comparison of model architectures, section 2 focuses on more practical issues. The final section, "A roadmap to commercial implementation," includes multiple points that are important to consider before choosing an architecture and beginning development. The questions identified here then motivate the exploratory studies collected in Chapter 3, which bridge this review and the main contribution chapters.

2.1 Methods in Computer Vision

Several review articles have been published on the topic of CV and AI that apply to plant pathology [88, 89, 90, 91]. High-quality works such as that by Weinstein *et al.* [89], which reviews the use of CV in animal ecology, are directly applicable to plant pathology owing to the flexibility of these techniques. What is missing from these works is a critical review and discussion of the latest and/or less conventional techniques in CV and a discussion of data acquisition and validation. Each of these reviews were published before or near the release of Detection Transformer (DETR) [92], Vision Transformer (ViT) [93], and ConvNeXT [94]. So naturally these recent landmark methods are not discussed. However, despite all being published after the release of Faster region-based convolutional neural network (Faster R-CNN) [95], ResNet [6], and You Only Look Once (YOLO) [96], only Xu *et al.* [91] mention any of these popular and high-performing architectures. Those being YOLO and region-based fully convolutional network (R-FCN) [97], an early predecessor of Faster R-CNN [91].

A recent survey [98] goes into great detail on the various facets of different attention mechanisms, which are integral to transformer architectures. While this work presents the bleeding edge of CV technology, it does not present the holistic, applied, and data-centric perspective provided here. Another paper aimed to develop CV models for the classification of cocoa beans, comparing the use of ResNet18, ResNet50, and support vector machines (SVMs) [99], while another recent review gives a high-level discussion of several CV studies in agriculture, covering hyperspectral imaging, the use of unpiloted aerial vehicles, and architectures as recent as ResNeXt [100, 101]. However, while the latter of these two papers presents a broad view of CV for plant pathology, providing strong links to many plant taxa, no mention is made by either [99] or [101] of architectures or techniques released after 2017. As such, the fusion of industry standard and bleeding edge methods in data acquisition, verification, and analysis presented here makes the present review unique among those listed above.

This review provides the reader with an in-depth understanding of CV for plant pathology and supports these previous works. In doing so, we focus on how best to adapt current methods to provide practical solutions for farmers, agronomists, and botanists without access to high-performance computational resources. While cocoa agriculture is used as a consistent example throughout, all methods discussed here are applicable across plant pathology and

agriculture, as well as related fields such as plant and animal ecology and forestry.

Ever since AlexNet was presented at the Conference on Neural Information Processing Systems in 2012, the field of CV has been dominated by convolutional neural networks (CNNs) [102]. While subsequent updates to CNN architectures have provided dramatic improvements over AlexNet [94], it is important to recognise that CNNs are not the only tools at our disposal. Previous work on cocoa disease has assessed the performance of SVMs, random forest regression, and other artificial neural networks to identify common diseases in cocoa from standard colour images, hereafter referred to as red, green, and blue (RGB) images [103]. Here it was shown that artificial neural networks are capable of identifying late-stage disease in RGB images of cocoa, but that training data set size is a limiting factor. Another study applied an SVM to perform pixel-wise identification of black pod rot (BPR) in cocoa [104]. The resulting algorithm showed an impressive ability to detect human-visible disease symptoms and, given the high computational efficiency of SVMs, it was able to run on low-powered hardware. Additionally, this model was trained on only 50 images, which is an extremely small training set in CV. However, no mention was made of the ability of these models to detect early disease development or non-human-visible symptoms, which will be a central focus of this review.

2.1.1 A Brief History of Machine Learning

Machine learning is the branch of AI concerned with building systems that improve their behaviour through exposure to data rather than through fully hand-written rules. Its intellectual roots lie in statistics, optimisation, information theory, and early cybernetics. In practical terms, the modern history most relevant to this thesis can be understood as a progression from simple pattern-recognition systems, to trainable multi-layer neural networks, to deep convolutional models, and then to attention-based and transformer-based architectures [105, 106, 102, 6, 107, 93].

The invention of back-propagation was especially important because it made it practical to train multi-layer neural networks by propagating an error signal backwards through the model and adjusting each weight according to its contribution to the final prediction error [105]. In image analysis, this development matured into the early CNN lineage, exemplified

by LeNet-style architectures, in which convolution and pooling allowed a model to learn spatially local features directly from pixels rather than relying entirely on handcrafted descriptors [106]. The deep-learning renaissance of the 2010s then accelerated when large labelled datasets, improved graphics processing units (GPUs), and more stable training recipes allowed AlexNet and its successors to outperform classical pipelines on large-scale visual benchmarks [102].

The basic learning paradigms that underpin the remainder of this thesis are summarised in Table 2.1. Supervised learning remains the dominant paradigm for disease classification because it learns directly from expert-labelled images. Unsupervised learning is useful when labels are absent and one instead wishes to cluster data, compress it, or model its latent structure. Semi-supervised learning occupies the middle ground and is especially important in plant pathology, where acquiring images is usually easier than producing reliable labels. The later chapters of this thesis make repeated use of supervised, semi-supervised, and generative approaches for exactly this reason.

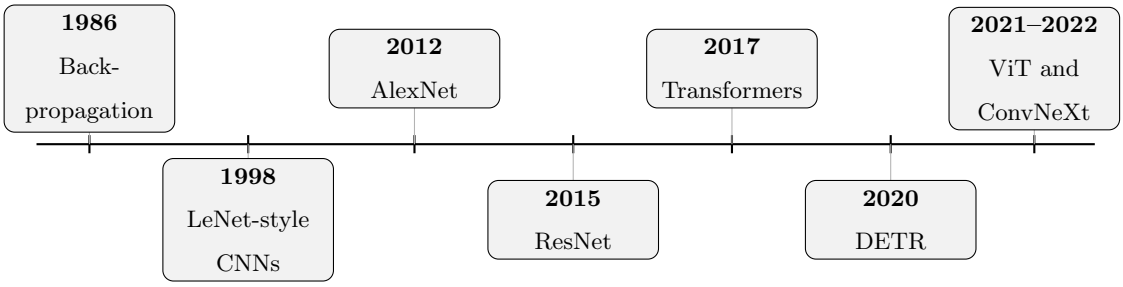


Figure 2.1: A simplified timeline of major machine-learning milestones relevant to the model families used later in this thesis. The emphasis here is not on completeness, but on the progression from trainable multilayer networks to modern convolutional and attention-based architectures.

Table 2.1: Core learning paradigms relevant to this thesis and their typical roles in plant-pathology workflows.

Paradigm	Labels required	Typical role in this thesis context
Supervised learning	Full labels	Learns a direct mapping from image to class label, for example disease diagnosis from expert-labelled images.
Unsupervised learning	None	Used to cluster data, reduce dimensionality, or learn latent structure when reliable labels are unavailable.
Semi-supervised learning	Some labels plus more unlabelled data	Useful when image collection is easier than expert diagnosis; enables a model to exploit harder or weakly labelled cases.
Generative modelling	Usually none or weak labels	Learns the distribution of the data itself, which can help with representation learning, anomaly detection, or data filtering.

2.1.2 Core Concepts in Neural Networks and Training

For readers coming from biology rather than computer science, it is useful to view a neural network as a long mathematical function composed of smaller functions called layers. In image analysis, the earliest layers usually learn simple local patterns such as edges, colour transitions, or texture. Deeper layers recombine these lower-level patterns into more abstract visual concepts, such as lesions, pod shape, or whole-tree structure. A convolution layer does this by sliding a set of learned filters across the image; an activation function such as rectified linear unit (ReLU) or Gaussian error linear unit (GELU) then introduces non-linearity so that the model can represent more complex decision boundaries than a simple linear classifier [108].

Pooling and normalisation layers play supporting but important roles. Pooling reduces spatial resolution and can make a model less sensitive to small local shifts, while adaptive pooling allows a network to accept variable input sizes and still produce a fixed-size representation. Normalisation layers instead stabilise the scale of internal activations. Batch normalisation

uses statistics from a mini-batch, while layer normalisation operates within a sample and is therefore often more stable for small batch sizes or transformer-style models [109, 110, 111]. These design choices matter because the models later compared in this thesis differ not only in depth and parameter count, but also in how they control optimisation stability and information flow.

Training proceeds iteratively. A model first performs a forward pass in which it maps an image to a predicted label or score. A loss function then measures how far that prediction is from the target. Back-propagation computes how each parameter contributed to that error, and an optimiser such as stochastic gradient descent or Adam then updates the parameters to reduce future error [105, 112]. Hyperparameters, by contrast, are values chosen by the researcher rather than learned directly by the model. Examples include learning rate, batch size, image input size, augmentation strength, dropout rate, and optimizer momentum terms. Hyperparameter tuning is therefore the process of searching for a training configuration that improves validation performance without simply memorising the training set.

The mechanics of this process are summarised in Figure 2.2. The central point is that the model is not learning one image at a time in isolation; rather, it is repeatedly adjusting millions of parameters in response to aggregate error patterns across many batches and many epochs. This is why choices in data curation, label quality, model size, and optimisation strategy can have as much impact as the architecture itself.

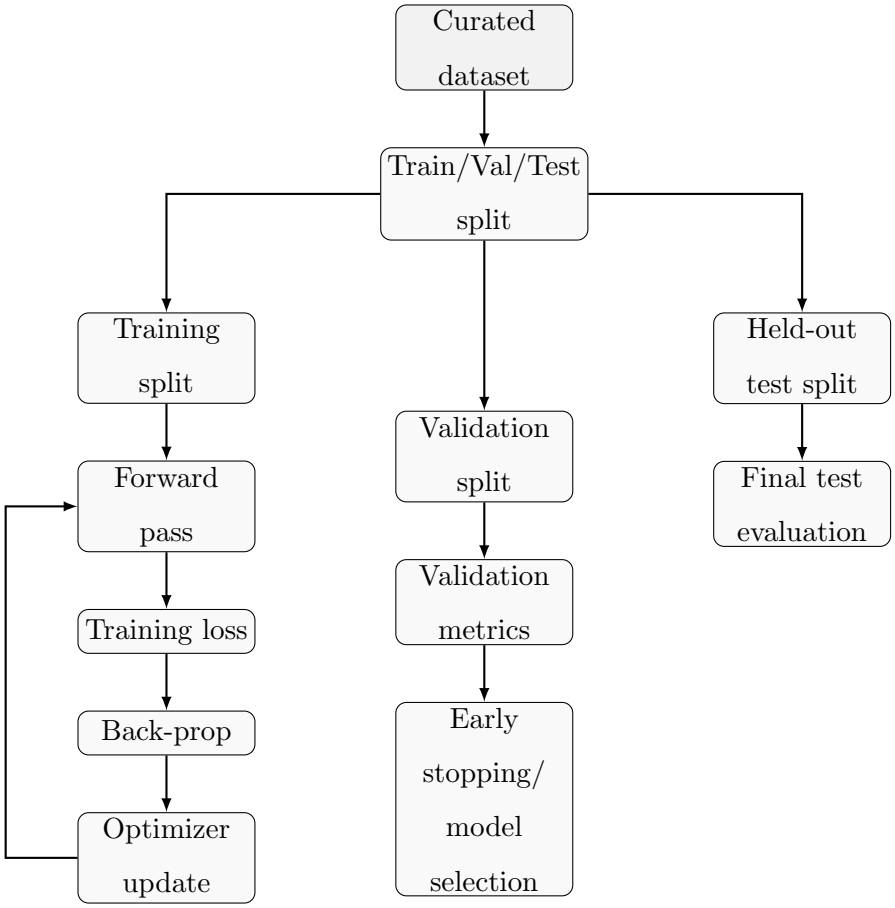


Figure 2.2: Simplified workflow for training an image model. The training split is used to update weights, validation is used for model selection and early stopping, and the held-out test split is reserved for final evaluation.

2.1.3 Hyperparameter Optimisation and Early Stopping

Once a model family has been chosen, much of the practical behaviour of the final system depends on hyperparameters rather than on learned weights alone. These include learning rate, batch size, image input size, optimiser settings, regularisation strength, architectural depth and width, kernel sizes, and normalisation momentum terms. In small or noisy biological datasets, such choices can change not only headline accuracy but also overfitting behaviour, stability of training, and runtime cost. For this reason, hyperparameter optimisation should be treated as part of model design rather than as a secondary implementation detail.

Grid search of hyperparameters is conceptually simple but becomes inefficient as the number of tunable variables grows. Random search is often more efficient when only a subset of

variables strongly affects performance. Bayesian optimisation goes a step further by fitting a surrogate model, often a Gaussian process, to the relationship between hyperparameters and a validation objective, and then proposing the next configuration using an acquisition rule such as expected improvement. This is especially attractive when each trial is expensive. Early stopping complements these search strategies by monitoring a validation loss or metric and halting training when continued optimisation no longer improves generalisation or worsens generalisation by overfitting to the training data. In later chapters, these tools are used not to maximise a benchmark score in isolation, but to identify configurations that balance fit, biologically informed model design choices, robustness, and compute efficiency.

2.1.4 Focal Loss and Hard-Example Reweighting

Cross-entropy loss remains the default objective for most classification tasks, but it can place too much emphasis on observations that are already easy for the model to classify correctly. Focal loss was introduced to reduce the influence of such easy examples and to focus training on hard or rare cases, especially in settings with severe class imbalance [113]. This idea is relevant beyond object detection because many applied plant-pathology datasets also contain observations that vary widely in difficulty, ambiguity, and prevalence.

Given the cross-entropy loss ($cross - entropyloss$(CE)) for an observation, the probability of correctly classifying the target class (pt) is defined as:

$$pt = e^{-\mathrm{CE}} \tag{2.1}$$

The focal loss ($focalloss$(FL)) for an individual example, weighted by α and focusing parameter γ, is then:

$$\mathrm{FL}(pt) = \alpha(1 - pt)^{\gamma}\mathrm{CE} \tag{2.2}$$

Finally, the batch-averaged focal loss for N observations is:

$$\mathrm{FL} = \frac{1}{N}\sum_{i=1}^{N} \alpha(1 - pt_i)^{\gamma} CE_i \tag{2.3}$$

In practice, focal loss is one example of a broader family of hard-example reweighting methods. Later in this thesis, we return to this idea by introducing a novel dynamic focal loss (DFLoss) that replaces the static emphasis on difficult predictions with a term informed by empirical difficulty during training.

2.1.5 A Brief History of Computer Vision

Computer vision is broader than modern deep learning and has roots in geometry, optics, signal processing, linear algebra, and probability. Long before contemporary neural networks, researchers studied how cameras form images, how three-dimensional scenes project onto a two-dimensional sensor, and how useful structure could be recovered by filtering an image mathematically. Classical computer vision therefore grew from operations such as convolution, gradient estimation, edge detection, corner detection, texture analysis, and feature matching [114].

This classical era typically relied on hand-engineered descriptors. A researcher would decide in advance which visual measurements might be informative, such as oriented edge histograms, keypoints, colour histograms, or shape descriptors, and then feed those features into a downstream classifier such as an SVM. Approaches such as the scale-invariant feature transform and histogram of oriented gradients were highly influential because they allowed systems to recognise local structure and object shape more robustly than raw pixels alone [115]. However, these methods also imposed a hard ceiling on what the model could learn: if an informative feature was not engineered into the pipeline, the downstream classifier could not discover it.

Deep learning changed this balance by collapsing feature engineering and classification into a single trainable pipeline. A modern CNN learns both the features and the classifier directly from data, while later vision transformers and detection transformers extend this idea by using attention mechanisms to model relationships across distant regions of an image. The progression most relevant to this thesis is therefore from image-processing mathematics, to handcrafted visual descriptors, to learned convolutional representations, and finally to modern architectures that combine local feature extraction with global context modelling.

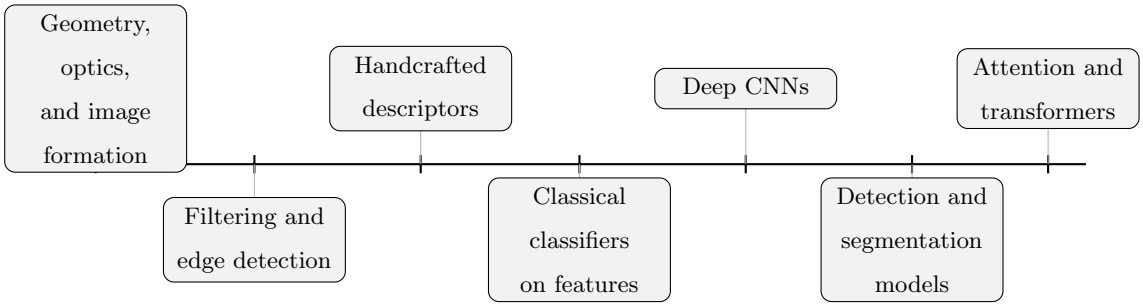


Figure 2.3: A simplified computer-vision timeline showing the progression from mathematical image analysis to learned visual representations. This broader history is important because many modern CV models still inherit ideas from classical filtering, multiscale analysis, and geometric reasoning.

2.1.6 Vision Transformers

In the early 2010s, transformers became the default for natural language processing [94], and they are now rapidly gaining popularity in vision-based tasks. Their central innovation is the attention mechanism, which allows a model to assign different weights to different tokens or regions according to estimated relevance. In practical terms, attention helps a model decide which parts of an image should influence a given prediction most strongly, while multi-head attention allows several such relations to be modelled in parallel [107, 98]. This ability to model long-range relationships is one reason transformers perform well in tasks where global context matters.

Pure transformer-based multilayer perceptrons, such as ViT [93], do away with the convolutional layers of a CNN. Instead, they subdivide and tokenise an image, give each token a positional embedding, and then pass all of these data to the multi-head attention mechanism of the network. The main drawbacks of such transformer-based models are that they require training data sets on the order of millions of images, and they lack the inductive biases of CNNs, such as translational equivariance [93]. In addition, the global structure of objects in an image must be learned from scratch, whereas this is maintained throughout a CNN. However, when pre-trained on a large data set and then fine-tuned on a more modest data set of tens of thousands of images, vision transformers can outcompete CNNs [93].

Although the requirement for vast training data sets may preclude the use of transformers for many plant pathology projects, there is a middle ground between the popular ResNet

architectures and transformer models. Taking inspiration from transformer designs, the highly competitive ResNet architectures were updated to produce a pure CNN that competes well with transformers in many tasks and is reported to outperform the original ResNets by about 3% accuracy on ImageNet [94]. This family of four models is named ConvNeXt and includes models of varying complexity from ConvNeXt tiny to ConvNeXt large. Additionally, ConvNeXt uses layer normalisation in place of batch normalisation. This modification could have important benefits for plant pathology projects, as discussed in the "Image, batch, and layer normalisation" section; however, as the ConvNeXt architectures are relatively large (ConvNeXt-tiny: 29 million parameters, ResNet18: 12 million parameters, ResNet50: 26 million parameters), these models too require large and/or complex training data sets to avoid overfitting, as well as more powerful hardware to run at inference than the smaller ResNets.

2.1.7 Object Detection and Semantic Segmentation

Bounding box object detection and semantic segmentation are processes by which objects of interest in an image are both classified and located in the image. In these tasks, either a box (bounding box object detection) or a polygon or "mask" (semantic segmentation) is drawn around the object of interest. The promise of these approaches for plant pathology motivated the first exploratory study reported later in Section 3.2.

Semantic segmentation and object detection could help in the accurate manual labelling of disease states in images. In simple image classification with a CNN, a model must learn what features, across the whole image, can be used as true markers of disease. However, annotation of training images with bounding boxes or segmentation masks may be used to focus the attention of the model, thus making training more efficient and reducing overfitting. This beneficial effect might be more pronounced with semantic segmentation than bounding boxes because the edges of a bounding box may extend beyond the edges of the leaf, pod, or tree in question and thus mislabel parts of neighbouring healthy plants. However, when comparing the ability of Faster R-CNN and Mask region-based convolutional neural network (Mask R-CNN) to detect human-visible signs of insect damage in sweet peppers (*Capsicum annuum L.*), Faster R-CNN was shown to have superior accuracy and mean av-

erage precision (mAP) [116]. Here, mAP is defined as the mean precision over all classes of the mean per-class precision, with a given intersection over union. These disparities in performance were contingent on which backbone model architecture (Inception v2, ResNet50, or ResNet101) [117] was used. When the more complex ResNet101 was used, Faster R-CNN and Mask R-CNN performed more similarly, although in this task Faster R-CNN performed best with the simpler architectures [94]. However, it should be noted that average precision is not directly comparable between bounding box detection and semantic segmentation models. This is for two reasons: (1) it is easier to achieve a given intersection over union with a bounding box as this task is less precise than segmentation, and (2) Mask R-CNN simply adds the ability to predict a mask in a box predicted by Faster R-CNN, so segmentation is additive in this case. As such, the results of [116] should be considered accordingly.

Object detection and semantic segmentation are typically performed using region-based convolutional neural network (R-CNN)-style models such as Faster R-CNN [95], Mask R-CNN [118], or YOLO [96]. However, these architectures have also been combined with other methods, such as SVMs, to confirm or deny the presence of an object in a proposed region [88]. For example, SVMs have been used in conjunction with Mask R-CNN in automated ML pipelines to identify defects in machined parts [119]. Additionally, when facing a classification problem with high intraclass variance, low interclass variance, and insufficient training examples, the application of SVMs to features learned by a CNN from ImageNet can improve results relative to a CNN alone [120]. This may prove useful in projects with few training images, or when classifying images of plant disease with similar characteristics, such as BPR in cocoa caused by *Phytophthora megakarya* or *Phytophthora palmivora*.

Furthermore, while *P. megakarya* and *P. palmivora* can be distinguished by eye, *Lasiodiplodia* species, of which three are known to infect cocoa, can present with identical morphological characteristics. This means that traditional classification techniques are insufficient and molecular identification techniques must be used in their place [121]. The development of CV technologies that can make such difficult distinctions would have important implications for all areas of agriculture and botany for two reasons. First, while *P. megakarya* and *P. palmivora* are managed in the same way, different species of *Lasiodiplodia* are not [122]. Thus, the failure of a model to distinguish between species of *Phytophthora* is not critical for effective disease management, but failure to distinguish between species of *Lasiodiplodia*

is. Second, cosmopolitan pathogens such as *Phytophthora spp.* and *Lasiodiplodia spp.* have extremely wide host ranges, infecting many commercially important crops. *Lasiodiplodia theoromae* alone attacks over 189 plant species across 60 families [123], while the growing list of described *Phytophthora* (aka "plant destroyer") species is currently 116 entries long [124].

Transformer-based object detection models such as DETR [92] are also now available and contend well with Faster R-CNN when trained on the huge Common Objects in Context (COCO) [125] benchmark data set. The key benefit of DETR is that it predicts bounding box coordinates directly, negating the need for the region proposal network of Faster R-CNN. Faster R-CNN's region proposal network has issues trying to identify overlapping objects because of the non-max suppression algorithm, which was removed from YOLO in version 3 [126]. However, DETR has problems detecting small objects and has a very long convergence time. These defects are resolved in later iterations such as Deformable DETR [127], the much improved DETR with improved de-noising anchor boxes (DINO-DETR) [128] and real-time Detection Transformer (RT-DETR) V2 [129], which incorporates several of the benefits of DINO-DETR, such as de-noising loss, while maintaining fast runtime speed.

In segmenting instances of nuclei in microscopy images, Mask R-CNN was compared with the U-Net architecture [130], which was designed for medical image segmentation. Here, the two techniques were shown to give similar mAP, F1 score (F1), and recall scores [131], although Mask R-CNN scored 0.812 for precision, while the U-Net scored only 0.68. A subsequent ensemble approach was then described, which shares the outputs of the two independently trained architectures to exploit the U-Net's purportedly superior F1 scores (+0.057), in tandem with Mask R-CNN's high mAP, precision, and recall. The ensemble model produced comparable, if slightly higher, mAP (+0.016), F1 (+0.056), and recall (+0.037) scores compared with Mask R-CNN, but the precision was 0.087 lower. Although the U-Net was reported to produce the best F1 score and the ensemble model produced the best mAP and recall, these improvements were slight. Additionally, F1 is calculated directly from precision and recall, so it seems counterintuitive that the U-Net approach could have the highest F1, yet the lowest precision and recall. The most noteworthy result here is the consistently superior precision of Mask R-CNN in this comparison and in another against YOLO [132, 126]. Additionally, in a study comparing the use of U-Net and Mask R-CNN to segment images

of pomegranate (*Punica granatum L.*) trees, Mask R-CNN outperformed the U-Net in both precision and recall by wide margins [133].

An alternative approach applied an SVM to perform pixel-wise classification to detect black pod rot in cocoa, with a human expert labelling the diseased pixels in training images [104]. Like semantic segmentation, this technique achieves the effect of providing the model with additional information on the location of disease symptoms in an image, relative to simple classification with a CNN. However, this imposes arbitrary physical boundaries around disease symptoms such as lesions and cankers, so the algorithm is unable to define for itself any symptoms that are not or cannot be identified with human vision. By using semantic segmentation with a CNN backbone, like in Mask R-CNN or DETR, to segment whole trees, these effects could be avoided. As in a typical CNN, the model would then be able to detect non-human-visible symptoms via feature learning and capture the effects of biological processes away from the site of infection. The feasibility of that trade-off is examined in Preliminary Investigation 1 (Section 3.2).

2.1.8 Variational Autoencoders for Outlier Detection

In addition to discriminative modelling, AI provides several powerful tools for generative modelling. Modelling with a generative deep neural network (DNN) can aid in gaining an intuitive understanding of the physical laws that led to the creation of the data to be modelled. An example of this is the use of artistic style transfer with generative adversarial networks (GANs) [134], where specific semantic features in an image can be isolated and utilised. Another popular deep generative model architecture is the variational autoencoder (VAE), which we will focus on here for the task of image data set filtering.

When working with autonomously collected data, for example from camera traps or web-scraping bots, the acquisition of vast quantities of data is often the easy part of creating a good training data set. Camera traps tend to produce a significant amount of uninformative data and the data from naive web-scraping bots can be badly contaminated with misclassified and irrelevant images; for example, a search for the keyword “Acer” will return many more images of laptops than it will Japanese maple trees, and a search for “black pod rot” will include many images of frosty pod rot (FPR), cherelle wilt, and insect damage. Therefore,

some level of human supervision is vital in curating training data, and the importance of consulting farmers and researchers in data collection and labelling cannot be overstated. However, manual labelling of a full data set can be extremely costly, and a potential method to offset some of this cost is said to be the use of VAEs for outlier detection.

A VAE is composed of two neural networks that are trained concurrently. The encoder network projects the image data to a smaller latent vector space, thus compressing it, and the decoder network predicts the original image from this compressed data as best it can.

Generative models tend to generalise to the real world much better than discriminative models, which aim to uncover correlative relationships between data and class labels [135]. However, deep generative models are typically considered excessive for classification problems, they often have higher bias [136] and are computationally expensive.

Previous works have successfully used VAEs for text classification [137, 138], data clustering [139, 140], anomaly detection [141], recommender systems [142], and dimensionality reduction [143]. There are also a limited number of published papers on the use of VAEs for anomaly detection with colour images [144].

Here, we consider two methods by which a VAE might be used to detect outlying data in collections of large colour images. To make this concrete, we use the forestry and arable images from Google and Bing (FAIGB) dataset, described in Section 2.2.2, because its web-scraped origin and the need to refine noisy web crawler results through automated filtering and manual review create exactly the kind of contamination problem that motivates these methods.

Method 1. Distribution of reconstruction loss

Having trained a VAE on only plant images, use this model to compress and decompress all images in the contaminated data set and record the reconstruction loss for each image. Plot the distribution of the loss values and record the most extreme high values as outliers. The assumption here is that the model should fail to reconstruct non-plant images well, as it should be naive to any images that do not show plants.

Method 2. Dimension reduction and clustering

Using the encoder network of a VAE that has been trained on the ImageNet data set, compress the images in the contaminated data set and record the values of the latent space for each image. Reduce the dimensions of the latent space further with principal component analysis (PCA), t-distributed stochastic neighbor embedding (t-SNE), or uniform manifold approximation and projection (UMAP) and plot these reduced data. Outliers and contaminant images may then be visually separated from the clean data.

Nouveau variational autoencoder (NVAE) is the product of an effort to carefully craft the encoding network architecture of a VAE, which appears to produce excellent results [145]. After training for just one epoch, this architecture can project large colour images onto a latent space and reconstruct them almost perfectly. However, if the aim of using NVAE is to compress image data, this architecture is not appropriate. This is because, using the recommended settings for the CelebA 64 data set, a 64 $\times$ 64 pixel version of the CelebFaces Attributes dataset [146], the latent space produced for an image with dimensions (3,224,224) is (100,224,224), *i.e.* more than 33 times larger than the original image. Following the NVAE authors' provided instructions to constrain the latent space to be as small as possible without excessively modifying the code, the latent space for this same size of image remains the same (100,224,224). This observation is corroborated in another study where the authors explain how NVAE first expands the data dimensions to a large number of latent spaces before pruning those spaces based on Kullback–Leibler (KL) divergence [147]. However, these authors go on to note that, in their use case, NVAE transformed images of size (3,32,32) to a latent space of size (16,16,128) without any subsequent downscaling. It is not surprising then that this architecture can reconstruct an image so well after just one training epoch, with no pre-trained weights, as the dimensionality of the data is expanded rather than compressed. Likewise, NVAE is not appropriate for identifying outliers by the distribution of reconstruction errors, as it can reconstruct any image almost perfectly.

The paucity of papers published on the subject of outlier detection in colour images with VAEs seems to be due to the inherent difficulty of this task. The high dimensions of such data and the large storage and GPU memory requirements that training these models on such data necessitates [148] have largely been resolved, although for many projects GPU memory availability will still preclude this technique. Thus far, the inability of the VAE ar-

chitecture to learn a compression algorithm for large colour images suggests a hard physical limitation that might not be overcome. Moreover, while [149] contest this argument, [150] argue comprehensively that generative models are not suitable for outlier detection by the reconstruction loss method described above, as these models tend to learn low-level statistics about data rather than high-level semantics. As such, they are often unable to differentiate between images that, to the human eye, are obviously different such as an image of a tree and an image of a wooden piano in front of floral wall paper. These concerns motivated Preliminary Investigation 2 (Section 3.3), where reconstruction-based filtering and a conservative semi-supervised alternative were compared on contaminated plant imagery.

2.1.9 Semi-Supervised Learning and Weak Supervision

Semi-supervised learning occupies the space between fully supervised and unsupervised learning by combining a smaller labelled set with a larger pool of unlabelled or weakly labelled data. Common families include wrapper or self-training methods, graph-based label propagation, and generative approaches that model the joint structure of inputs and labels [151, 152]. The appeal of these methods in plant pathology is obvious: collecting images is usually easier than obtaining reliable expert labels, especially when symptoms are subtle or diagnostically ambiguous.

Wrapper or self-training methods are conceptually simple. A model is trained on labelled data, used to predict labels for additional samples, and then retrained on a selected subset of those predictions. Their main strength is practical scalability, but their main risk is confirmation bias: if incorrect pseudo-labels are admitted too early, the model can reinforce its own errors and progressively contaminate the training set. Weak supervision pushes this idea further by accepting noisy labels from web-scale scraping or heuristic rules. Such strategies can be effective for large-scale pre-training, but they also make model behaviour harder to audit because an improvement in benchmark accuracy does not guarantee that the learned features are biologically meaningful or robust outside the training distribution.

Uncertainty estimation is therefore central to safe semi-supervised learning. Classical mixture models and graph-based methods often express uncertainty or similarity structure more naturally, whereas deep classifiers typically rely on softmax confidence, which is known to

be imperfectly calibrated under distribution shift [153, 154]. In practice, conservative semi-supervised workflows for high-stakes biological classification often combine thresholding with staged relabelling, human review, or difficulty-aware admission criteria. The later chapters adopt exactly this kind of conservative strategy rather than assuming that high confidence alone is sufficient.

2.1.10 Evolutionary Algorithms

The field of CV is currently dominated by handcrafted DNNs with fixed topologies. However, the seldom-used techniques of evolved neural networks have real potential in plant pathology. In an evolutionary algorithm, one typically maintains a population of candidate models, evaluates each model with a fitness function, and then generates the next population by selection, mutation, and sometimes crossover. In this setting, the fitness function can reward not only predictive performance, but also compactness, sparsity, or runtime efficiency. Evolutionary methods are therefore attractive when the goal is not simply to maximise accuracy, but to discover an architecture that is appropriately scaled to the deployment setting.

From an optimisation perspective, neuroevolution differs from ordinary back-propagation because it searches over model designs rather than only adjusting the weights within a fixed design. This can be particularly helpful when the most important choices are discrete, such as how many blocks to use, which normalisation layer to apply, whether to include skip connections, or how wide a stage of the network should be. Gradient descent therefore learns *within* an architecture, whereas evolutionary search can help decide *which* architecture should exist in the first place.

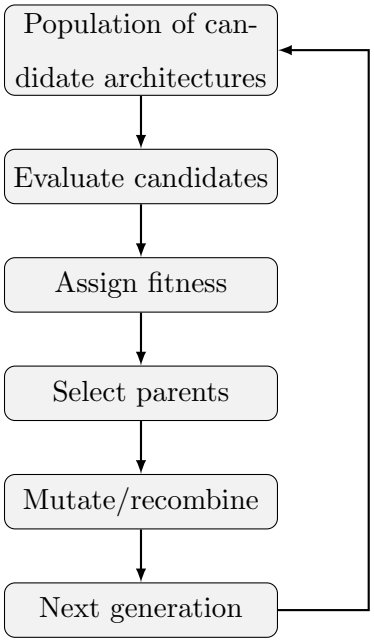


Figure 2.4: Conceptual workflow of evolutionary search for neural-network architectures. Candidate architectures are evaluated, assigned a fitness score, selected, and modified by mutation or recombination to form the next generation.

Computational efficiency at inference and improved ability to generalise are of paramount importance to models developed for plant pathology in the field. This is because such models must be able to cope with complex and highly variable symptoms and backgrounds, and often must run on low-powered hardware. Growing neural networks take far longer to train or search than those with fixed topologies, but this is of minor concern given efficient parallelisation and the availability of modern compute resources during development. By contrast, the hardware available to farmers in low-income sectors, such as cocoa, cassava (*Manihot esculenta Crantz*), or coffee (*Coffea spp. L.*) cultivation, is restricting. This restriction means that producing a model that is optimised for runtime speed at inference is a vital factor, and growing neural networks with evolutionary algorithms may be an ideal way to achieve this.

Evolving neural networks have been shown to be highly effective in producing neural networks with a high degree of modularity [155]. This increased modularity is said to be the result of applying a cost to the number of connections, which both reduces computational cost and promotes evolvability as the sharing of modular units between parents is made simpler. It is also said that such modularity helps these models to generalise better as each modular unit is capable of independent generalisation [156]. With evolutionary algorithms, one can also

Table 2.2: Test results of three architectures trained on two datasets to show an indirect comparison. ResNet18 was trained only on ImageNet with the top one and top five classification accuracies shown. EVOCNN was trained only on Fashion MNIST with the percent error shown. VGG16 was trained on both datasets. Results were taken from [160] and [162]. Number of parameters for VGG16 was mis-reported by [160] as 26 million.

Architecture	Top 1 acc.	Top 5 acc.	Error (%)	Parameters
ResNet18	69.758	89.078	–	11.7M
VGG16	**71.59**	**90.38**	13.78	138M*
EVOCNN	–	–	**7.28**	**6.52M**

promote diverse populations of networks with techniques such as niching [157], and use of non-elitism strategies can allow for the simultaneous exploration of fitness valleys and local optima without getting stuck there [158]. While elitism follows the biologically implausible assumption that the fittest individual/network will always survive to reproduce, non-elitism allows weaker individuals to explore fitness valleys, which may lead them to undiscovered maxima.

While a direct comparison of evolved neural networks with popular CNN architectures could not be found, table 2.2 shows an indirect comparison between a recent method for evolving neural networks (EVOCNN) and two popular CNNs, ResNet18 and VGG16; EVOCNN appears to perform very well in this comparison. However, the error rate for these models was calculated when trained on the Fashion-MNIST data set, while the top 1 and top 5 accuracy was produced using ImageNet. Fashion-MNIST, which is composed of 28 x 28-pixel greyscale images of clothing [159], is not a challenging proposition for modern CNNs and is not reflective of real-world plant pathology problems. Additionally, it should be noted that, in the EVOCNN paper [160], the number of parameters of VGG16 is misreported as 26 million, rather than the 138 million listed in the Torchvision documentation [161]. This suggests that VGG16 would have massively overfit to the Fashion-MNIST data, making this an inappropriate comparison. However, EVOCNN does offer a very low error rate on this simpler problem and has a very low number of parameters when compared with other modern architectures (Tables 2.2 and 2.3). Overall, it seems that evolved neural networks are not yet ready to tackle the more difficult problems in plant pathology, so more work is required in this area.

2.1.11 Architecture Comparison and Recommendations

The field of CV has produced a numerous and diverse set of architectures, each with unique strengths and weaknesses. Here, we will compare these architectures, focusing on their application in image classification, object detection, and semantic segmentation. Table 2.4 and table 2.5 give a detailed breakdown of the pros and cons of each of these architectures, as well as the number of trainable parameters, which acts as a proxy for model complexity, and the number of giga-floating point operations (giga floating-point operations per second (GFLOPS)), which gives a sense of computation cost of running inference with these architectures.

Table 2.3: High-level summary of the main model families and representative approaches discussed in this chapter. These families either appear directly in later comparisons or motivate methodological choices made in the subsequent empirical chapters.

Family	**Examples**	**Main use**	**Strengths and limitations**
Compact CNNs	ResNet18, PhytNet	Image classification	Efficient and well suited to modest datasets, but may miss broader context if scaled too conservatively.
Scaled CNNs	EfficientNet, ConvNeXt	Image classification	Often highly accurate, but larger variants can overfit small datasets and increase runtime cost.
Generative models	VAE, NVAE	Representation learning, outlier filtering	Useful for modelling data structure and some anomaly-detection tasks, but indirect and often computationally expensive for diagnosis.
Two-stage detectors	Faster R-CNN, Mask R-CNN	Detection and instance segmentation	Usually strong at localisation and precision, but comparatively heavy at training and inference.
One-stage detectors	YOLO	Real-time detection	Very fast and easy to deploy, but can underperform on small or visually subtle targets.
Transformer detectors	DETR, RT-DETR	Detection	Model global relationships cleanly and avoid hand-designed proposal stages, but often need more data and compute.
Evolutionary design	EVOCNN, neuroevolution	Architecture search	Can optimise jointly for accuracy and efficiency, but search is slow and current evidence in plant pathology is still limited.

Table 2.4: Pros and cons of popular model architectures for image classification. Ranges represent the smallest and largest off-the-shelf versions available. Trainable parameters (M) and GFLOPS were obtained from the PyTorch documentation.

Architecture	**n param.**	**GFLOPS**	**Pros & Cons**
ResNet (2015)	12–60M	1.8–11.5	**Pros:** ResNet18 is small and computationally efficient; suitable for modest datasets; widely used and tested.
			Cons: Batch normalisation can be unstable with small batch sizes.
EfficientNet V2 (2019)	22–119M	8.4–56.1	**Pros:** Scales depth/width/resolution with a single coefficient.
			Cons: Scaling often requires config/source edits; can overfit despite high test scores.
ConvNeXt (2022)	29–198M	4.5–34.3	**Pros:** Strong reported performance; modern components (GELU, stochastic depth, layer norm); scales via block settings.
			Cons: Off-the-shelf variants may be too large for small datasets; ONNX conversion may be problematic.
ViT (2021)	87–634M	17.6–1016	**Pros:** With large-scale pretraining, can slightly outperform very deep CNNs.
			Cons: Data-hungry; expensive training and inference.

Table 2.5: Pros and cons of popular model architectures for object detection and semantic segmentation. Ranges represent the smallest and largest off-the-shelf versions available. Values for Faster R-CNN and Mask R-CNN were obtained from the PyTorch documentation; YOLO and DETR parameters/GFLOPS were calculated for an input size of 224×224 pixels.

Architecture	**n param.**	**GFLOPS**	**Pros & Cons**
Faster R-CNN (2015)	44M	280.4	**Pros:** Often higher mAP than YOLO; performs better on small objects.
			Cons: More computationally expensive than YOLO; can struggle in crowded scenes depending on settings.
Mask R-CNN (2017)	46M	333.6	**Pros:** Extends Faster R-CNN with instance segmentation masks; strong baseline for segmentation tasks.
			Cons: Computationally expensive; heavier than Faster R-CNN.
YOLO (2016)	7M	1.01	**Pros:** Extremely fast at inference; fast to train; easy to implement.
			Cons: Often weaker on small objects; accuracy can lag two-stage detectors depending on variant/training.
DETR (2021)	40M	11.2	**Pros:** Removes region proposals and NMS; handles overlap well; transformer approach shows promise; faster inference than Faster R-CNN.
			Cons: Expensive training; slow convergence; data-hungry; can be challenging to implement; often benefits from large batch sizes for stability.

2.1.12 Image Classification Architectures

ResNet introduced the concept of skip connections, enabling the training of much deeper models. Despite its age, ResNet remains a strong competitor, and ResNet18 is probably still the best choice for most small projects with fewer training examples. EfficientNetV2 [163] is more computationally demanding than equivalent ResNet and ConvNeXT variants, and while it tends to yield high accuracy on large data sets [93, 94], we found that it is prone to overfitting, making it a less favourable choice. The key innovation of EfficientNet was to allow the depth, width, and resolution of the model to be scaled by adjusting a single coefficient [163]. However, in practice this requires editing the source code, thus rendering such adjustments less than convenient. ConvNeXT is an updated version of ResNet, incorporating several modern features. Unlike EfficientNet, ConvNeXT is easy to scale, making it a promising choice for medium- to large-scale applications, for which it has been shown to give superior performance to ResNet and VIT [94]. As the first transformer to perform favourably against CNNs for image classification, VIT represents a significant milestone. However, image classification may not be the optimal use case for transformer architectures, and at present ConvNeXT outperforms VIT while requiring less data for training and being less computationally expensive [93].

2.1.13 Object Detection and Semantic Segmentation Architectures

Although more complex than YOLO, and arguably DETR, Faster R-CNN delivers excellent results and requires only modest resources for training. For most object-detection use cases in plant pathology, Faster R-CNN will be the optimal choice. Mask R-CNN extends Faster R-CNN by adding the ability to predict a mask in a bounding box, enhancing its utility for semantic segmentation tasks. YOLO is most suitable for real-time object detection but offers lower precision than Faster R-CNN. It is not suitable for use in plant pathology unless inference time is of primary concern. DETR and Deformable DETR present a novel approach to object detection and offer competitive results [127]. However, implementing these architectures can be difficult and they require substantial GPU video random-access memory (VRAM) for training.

The choice of CV model architecture for a given project depends on a variety of factors, including data set size, signal-to-noise ratio, computational resources, mode of deployment, and accuracy requirements. However, at present, for most use cases in plant pathology, ResNet18, ConvNeXT tiny, or Faster R-CNN will yield the best results while minimising computational cost, risk of overfitting, and the financial cost of training.

2.1.14 Image, Batch, and Layer Normalisation

In a comparison of EVAL-COVID [164] with other strong competitors like EVOCNN to detect COVID-19 with evolved CNNs, it was shown that the overuse of batch normalisation (batch normalization (BN)) can be deleterious to the training of DNNs for disease diagnosis. While BN often improves the training time of CNNs and can negate the need for small learning rates and dropout [117], those benefits must be weighed against possible shifts in model behaviour on subtle disease features. This tension motivated Preliminary Investigation 3 (Section 3.4), which tests how image, batch, and layer normalisation behave on cocoa disease imagery.

Several state-of-the-art generative models now omit BN entirely, while others replace it with weight normalisation or focus on fine-tuning the momentum hyperparameter of BN layers [145]. Replacing BN in a standard ResNet with the alternative layer normalisation (layer normalization (LN)) can also worsen performance [111], whereas the ConvNeXt family was designed around LN from the outset and trains well with that choice [94]. The BN momentum hyperparameter is a fixed weight applied to the running mean and variance calculations that are tracked during training and used during inference. Thus, adjusting the BN momentum will not alter the direct effect of the training update itself [145], but it can affect how activations are normalised at evaluation time. This makes it a reasonable hyperparameter to optimise on a per-dataset basis and motivated Preliminary Investigation 4 (Section 3.5).

Taken together, the literature suggests that unnecessary image normalisation should be treated cautiously in plant disease detection, and that the choice and tuning of normalisation layers are data-dependent rather than universal defaults. Those lessons are taken forward in the design decisions of the later contribution chapters.

2.2 Data Acquisition and Model Testing

In this section, we review various interdisciplinary methods available for gathering a training data set and developing a suitable model. While the previous section concerned the theory of ML in CV, this section will focus on practicalities

2.2.1 Obtaining the Required Training Dataset

Training an image classifier to high accuracy in a controlled laboratory environment is often a trivial task. However, such a model may perform poorly when presented with the challenges of the real world [165]. For example, after training a leaf-disease classifier on images taken in the field, the model performed with around 68% accuracy when tested against images taken in the lab [166]. However, when trained in the lab and tested in the field, the same model architecture performed with about 33% accuracy. This effect is likely due to the plain white background of the lab images causing the model to generalise poorly to real-world applications. This exemplifies the importance of curating a realistic, high-quality training data set. By naively training and releasing models that are trained on publicly available data sets, we risk exacerbating the problems of disease misclassification. At low frequencies, the effect of mislabelled, misleading, or uninformative data will have a limited effect on the performance of a neural network. This feature of neural networks is largely an artefact of batch gradient descent and the learning rate [167], which act to greatly buffer the effect of infrequent misclassifications in the training data. At higher frequencies, these sources of error can have more serious consequences. The most obvious solution to this problem is to carefully curate, label, and annotate the training data. However, errors resulting from misclassification can be challenging to eradicate. For example, FPR, BPR, and witches' broom disease (WBD) in cocoa can all present with black or brown lesions on the pod, and both frosty pod root and black pod root can both coat a pod in white mycelium. This means that without sufficient training in plant pathology or access to diagnostic tests, one could easily mislabel these diseases. This problem can be solved by two means, which should be used in tandem: (1) by paying careful attention to detail and applying detailed knowledge of the pathogen in question, and (2) by using tools and techniques from molecular biology and spectroscopy to better inform model development and subsequent disease detection. Such

techniques/tools include deoxyribonucleic acid (DNA) sequencing, real-time quantitative polymerase chain reaction (RT-qPCR), loop-mediated isothermal amplification (LAMP), MultispeQ (PhotosynQ, East Lansing, Michigan, United States of America (USA)), and hyperspectral imaging (HSI).

2.2.2 The FAIGB Dataset

A recurring example in this thesis is the FAIGB dataset, which was created for this thesis work. FAIGB was assembled with a custom Python web-scraping pipeline adapted from an open-source Google Images scraper and extended to automate image collection for multiple forestry and arable taxa. Species names and disease metadata were defined in a CSV file, from which healthy and diseased search queries were generated automatically. Healthy queries used taxonomic names, whereas diseased queries incorporated pathogen or symptom terms. Selenium-driven browser automation was then used to interact with Google Images, and Bing where available, collect image URLs, and execute each query before downloading.

Downloaded files were retrieved with HTTP requests and processed with the Python Imaging Library. Images that failed to download or parse were discarded, images outside basic resolution bounds were filtered out, and unsupported formats were converted to RGB where necessary. The resulting files were organised into class-specific directories corresponding to healthy and diseased categories. Although the candidate images and provisional labels originated from search-query context, the FAIGB dataset used in this thesis was subsequently refined with the automated filtering procedures developed in this thesis and then manually reviewed, so it should be regarded as a rigorously curated dataset rather than as a merely weakly supervised benchmark.

In its curated form used throughout this thesis, FAIGB contained 31,225 images spanning 58 taxa and 116 healthy/diseased classes, of which 25,567 were healthy and 5,658 were diseased. The dataset spans forestry taxa such as *Abies*, *Acer*, *Betula*, Cupressaceae, Pinaceae, *Quercus*, *Salix*, and *Ulmus*, alongside crop taxa such as apple, banana, cassava, cocoa, cucumber, grape, maize, rice, soybean, strawberry, sugarcane, tomato, watermelon, and wheat. The class distribution is deliberately broad and strongly imbalanced, which makes the dataset useful for studying web-scraped data curation, outlier filtering, and future hard-example

Table 2.6: Summary of the FAIGB dataset used for web-scraped plant-image filtering and later as a source of non-cocoa negatives.

Property	Value
Source	Google Images and Bing
Collection pipeline	Custom Python scraper with Selenium-driven query execution, URL aggregation, and deduplication
Taxonomic scope	58 forestry and arable taxa
Class structure	116 rigorously curated healthy/diseased classes
Image total	31,225 images
Class balance	25,567 healthy and 5,658 diseased images
Cocoa content	659 cocoa images (226 diseased, 433 healthy)
Quality control	Failed-download removal, basic resolution filtering, format handling, automated filtering, manual review, and RGB conversion where needed
Label provenance	Candidate labels derived from search queries and retained after automated filtering and manual review

reweighting.

FAIGB is used in two ways in this thesis. First, it provides the concrete web-scraped plant dataset used in the outlier-detection and semi-supervised filtering examples discussed earlier in this chapter. Second, after cocoa images were removed, it provided the source pool from which the non-cocoa class was sampled in Chapter 6. This distinction matters because FAIGB itself contains 659 cocoa images, so the later non-cocoa subset was drawn only after those images had been excluded.

2.2.3 Off-the-Shelf Datasets Used in Later Chapters

In addition to custom datasets, later chapters also rely on a small number of widely used public datasets. ImageNet is a large-scale hierarchical natural-image dataset containing over one million images organised across one thousand object categories [168]. In this thesis, ImageNet is relevant for two reasons. First, it is the canonical source of off-the-shelf pre-trained weights against which custom plant-pathology training choices are discussed in later chapters. Second, in the data-filtering example above, a random subset of ImageNet was used as a source of non-plant negatives when training the binary Outlier-NoneOutlier classifier. This makes ImageNet useful as a generic contrast set, even though its object categories are largely irrelevant to specialised cocoa disease recognition.

A second public dataset used later is the “Enfermedades cacao” dataset [169]. This is a

small cocoa-pod disease dataset released for object-detection experiments and composed of annotated images of healthy pods together with Phytophthora and Monilia symptoms collected in Colombia. In this thesis it was not treated as a standalone benchmark. Instead, images from its Vivo subset were incorporated into the cocoa dataset used in Chapter 6 to increase diversity in image source, acquisition device, and symptom presentation. This is why the later class-frequency table distinguishes a Vivo source from the images gathered directly in the field, from the web, or from FAIGB.

2.2.4 Train, Validation, and Test Splits

Once a dataset has been curated, the next critical step is deciding how it will be partitioned. The training split is used to fit model parameters, the validation split is used to tune hyper-parameters and detect overfitting during development, and the test split is reserved for the final assessment only. In principle this sounds simple, but in biological datasets the most common failure mode is data leakage. Leakage occurs when near-duplicate images, images from the same plant, or images captured under almost identical conditions appear in both the training and evaluation sets. A model can then appear to perform very well while in fact having learned only the idiosyncrasies of a particular site, camera, or sampling event.

For this reason, split strategy must reflect the biological question being asked. If the aim is to generalise across farms, then images from the same farm should not be distributed casually across train and test sets. If the aim is to detect disease across time, then images from the same progression sequence should not leak between sets. In small datasets, simple hold-out testing can be unstable because the estimate depends heavily on which images happen to fall in the test split. Cross-validation therefore becomes useful because it repeats the train-evaluate cycle across multiple folds and yields a distribution, rather than a single point estimate, of performance [170].

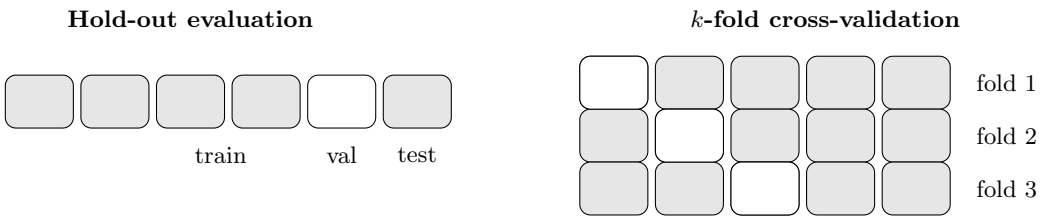


Figure 2.5: Two common validation strategies. Left: a conventional hold-out split. Right: cross-validation, where a different subset is held out in each fold. Cross-validation is particularly useful for smaller datasets because it produces a distribution of performance values and reduces dependence on a single arbitrary split.

2.2.5 Tools from Molecular Biology

DNA and ribonucleic acid (RNA) are the molecular record and working copies of biological information. From the perspective of this thesis, the key point is that disease symptoms are the visible consequence of underlying molecular events: pathogen growth, host defence responses, tissue degradation, and physiological stress. Molecular assays are therefore useful because they can confirm whether a visually subtle or ambiguous symptom really is associated with a given pathogen. They also allow a researcher to detect infection before severe human-visible symptoms emerge, which is especially valuable when one wishes to understand what a model may be detecting at the earliest stages of disease.

Polymerase chain reaction (PCR), quantitative polymerase chain reaction (qPCR), nucleotide sequencing, and LAMP should therefore be viewed as complementary tools rather than alternatives to CV. CV is attractive because it is fast, scalable, and deployable on farms. Molecular biology is attractive because it is highly specific. In combination, they make it possible to build datasets in which the labels are biologically defensible rather than based only on gross symptom appearance. This is particularly important in cocoa because lesions caused by different pathogens can overlap visually, while the same pathogen can express differently across host variety, climate, and stage of infection.

At a practical level, PCR is a targeted copying process that amplifies a chosen fragment of genetic material, qPCR measures that amplification as it happens, and sequencing reads the order of nucleotides themselves. These methods therefore answer slightly different questions: whether a target is present, how much of it is present, and exactly which organism or variant

it belongs to.

DNA sequencing is now commonly used for the identification of cryptic species [171, 172] and plant pathogens [173] and is an invaluable tool. Once sequenced, reads can be used to search previously categorised sequences with the Basic Local Alignment Search Tool (BLAST) from the National Center for Biotechnology Information [174] to identify a sample at the level of species or other taxonomic grouping. However, if we know which pathogen(s) we expect to detect, sequencing the whole genome is excessive. Rather, we can use loci such as the internal transcribed spacer (ITS) region of the nuclear ribosomal RNA genes, which are both highly conserved across taxa and highly variable between species. Such regions of the genome can be amplified using PCR or LAMP to detect a pathogen or identify it with relatively low cost and high accuracy. The ITS region is often sequenced on its own for near–species level identification or in concert with other loci for better specificity [175]. Such work with ITS is now ubiquitous in the molecular study of fungal ecology and phylogeny, while previous techniques relied on the morphology of fruiting bodies for identification [175]. qPCR is used to detect asymptomatic disease across the agricultural industry [176]. Traditionally, PCR was unsuitable for portable operations or use in the field [177]. However, rapid RT-qPCR in the field is now possible [178]. RT-qPCR can also be used to quantify the relative levels of a pathogen in plants [179]. Information from such analyses could be extremely informative when fine-tuning and assessing the performance of the models discussed here.

LAMP can be used in place of qPCR and has four key benefits: (1) it is considerably cheaper (£211 or $256 USD for 100 samples) because a thermal cycler is not required; (2) it is fast; (3) the reagents do not need to be refrigerated; and (4) like real-time PCR, there is potential for it to be used in the field. Like qPCR, LAMP can be used to quantify the relative amount of DNA present, as well as simply for detection. If detection is the only goal, colour- or turbidity-based methods can be used to detect DNA presence by visual inspection. A drawback of this method is that any pre-existing PCR primers cannot be used. This is because, while PCR primers are designed to amplify a specific region of complementary DNA, LAMP primers bind to multiple regions of the target DNA in a way that allows for the simultaneous amplification of multiple regions of the DNA.

While universal PCR primers for the ITS region exist, it may be necessary to design LAMP

primers or species-specific PCR primers for ITS or other regions. For a detailed discussion of the use of ITS amplification in fungal ecology and the potential pitfalls of specific ITS primer design, see [175].

If novel primers are to be designed, the region of interest must first be sequenced, and if we aim to identify a currently unknown pathogen with BLAST, all of the DNA in a sample must be sequenced. Sequencing with the Oxford Nanopore Technology (Oxford, United Kingdom) MinION platform can be an ideal tool for this purpose, offering multiple features: (1) The Oxford Nanopore Technology field sequencing and library preparation kit allows for sequencing in the field, immediately after tissue samples are gathered, which eliminates the need for cold-chain storage to avoid sample degradation, (2) It allows for high-quality sequencing in countries where Illumina (San Diego, California, USA) sequencing is not available, (3) It is slightly cheaper than using the Illumina platform, and (4) The long read length eliminates amplification bias [180]. The avoidance of amplification bias is important for gene expression quantification, which is relevant to the discussion in section 2.2.9 . On the other hand, the MinION 1B requires a high-spec computer and, at a cost of £98 or $119 USD per sample excluding library preparation, the use of this platform also remains too expensive for many projects.

2.2.6 Spectroscopy and Hyperspectral Imaging

Spectroscopy concerns the interaction between matter and electromagnetic radiation. In plants, the reflected spectrum is shaped by pigments, water content, cellular structure, wax layers, and tissue damage. Disease can alter all of these factors. A diseased pod or leaf may therefore reflect light differently from a healthy one even when the difference is subtle or outside the wavelengths that humans can perceive. This is one reason spectroscopy is so relevant to plant pathology: it can reveal physiological disturbance before it becomes visually obvious in a standard photograph.

Although not capable of specific disease diagnosis on its own, the MultispeQ is an important low-cost tool to consider in the context of disease detection in the absence of visible symptoms. This handheld plant phenotyping device can be used to indicate the non-specific presence of plant disease at an extremely low cost [181]. The MultispeQ operates similarly

to photo spectroscopy and measures environmental conditions such as light intensity, temperature, and humidity. It can also be used to measure photosystem II quantum yield, which is an indicator of plant health, and to detect non-photochemical exciton quenching, which has been shown to have a significant negative correlation with disease index [181].

A highly informative technique that we can utilise in the prediction of plant disease with CV is to sample more continuously from the electromagnetic spectrum with HSI. As with the MultispeQ, HSI enables the detection of changes in the chemical composition of biological tissue in terms of conditions such as ripeness or disease status change [182]. The term "spectral signature" is used to describe the pattern of electromagnetic radiation reflected by a subject. However, particularly in the case of biology, the term signature is misleading as biological samples often have highly heterogeneous reflectance spectra [182]. All of the above-mentioned CV studies applied ML techniques to RGB images. RGB images capture three discrete bands of the visible spectrum from 400–700 nm. Black-and-white digital images have two spatial dimensions and a single dimension that describes the darkness of each pixel on a scale of 0–255, whereas RGB images have three colour dimensions represented by values between 0–255, each describing the intensity of red, green, or blue light. Hyperspectral images, however, store a more complete reflectance spectrum for each pixel, while also maintaining spatial relationships. The spectral range of these images can be as wide as 300–2500 nm [183].

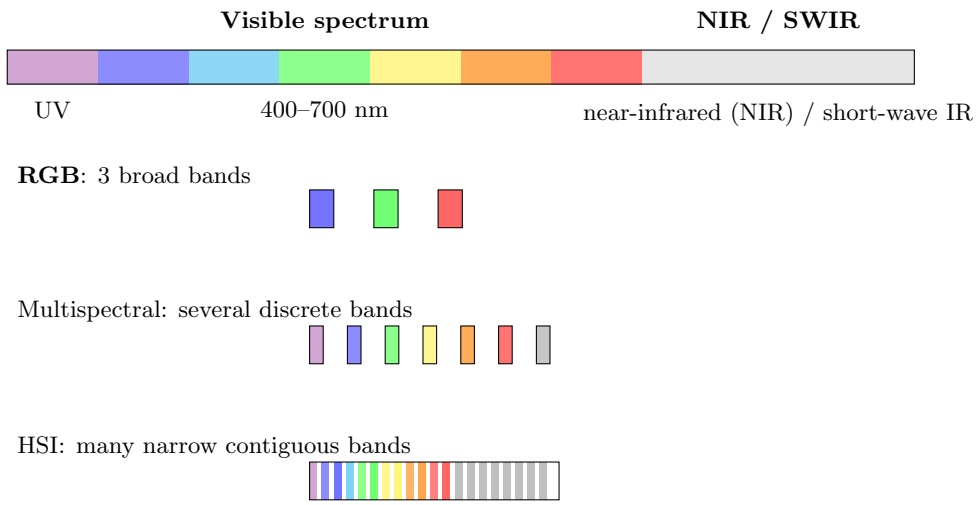


Figure 2.6: Conceptual comparison of spectral sampling by standard RGB, multispectral, and hyperspectral sensing. Standard photographs capture three broad visible bands, multispectral systems sample a small number of discrete wavelength bands, and hyperspectral sensing samples many narrow adjacent bands. This denser spectral sampling can capture subtler spectral differences relevant to plant physiology and disease detection.

Although the applications of hyperspectral photography have long been explored by NASA, this technology is only now becoming affordable for use in industries such as agriculture. Commercially available cameras capable of capturing data from the 300–2500 nm range remain expensive, and more typically the cameras used only sample 330–1100 nm (table 2.7). Despite the reduced spectral range of the cheaper cameras, they still provide orders of magnitude more data than RGB cameras, although much of this data is highly correlated.

The uptake of HSI has recently exploded in a host of fields including archaeology, art conservation, food safety, medicine, and crime scene investigation [184]. Typical applications of HSI in agriculture include the estimation of yield [185, 186], assessment of vigour [187], remote weed identification [188], nutrient status [189], and disease monitoring [190].

The analysis of HSI data presents problems that are familiar to ML engineers and nowadays are solved routinely. These problems include the large size of HSI hypercubes, high dimensionality, high intra-class variability, and high correlation between spectral bands. Many approaches have been taken to analyse these data and, for a long time, SVMs were the most widely used [191]. Today, DNNs are commonly used to analyse these data as they are particularly well suited to this task. DNNs are able to isolate hidden and complex data structures,

can utilise a great variety of data types, are flexible in their architectures and the complexity of the mathematical functions they can apply, and are ideally suited to distributed computing [192]. As such, with the addition of dimension-reduction techniques such as PCAs [191], the analysis of HSI data with DNNs, although more computationally demanding, becomes little more complex than such analyses of RGB image data.

While the field of CV is advancing at a rapid pace, so too are the fields of molecular biology and spectroscopy. The use of tools and knowledge from these fields will allow projects of various budgets to go beyond the simple application of CNNs to RGB images and, in doing so, model disease in greater detail and with tangible biological explications of model behaviour.

Table 2.7: Specifications and use cases for the hyperspectral cameras used in the following studies [188, 187, 185, 190, 186, 189]. Manufacturer locations are: Resonon (Bozeman, Montana, USA), Headwall Photonics (Bolton, Massachusetts, USA), Specim (Oulu, Finland), BaySpec (San Jose, California, USA).

Make/model	**Task**	**Spectral range (nm)**	**Spectral bands**	**Spectral resolution (nm)**
Resonon Pika II Vis–NIR	Mango tree yield estimation	390–890	244	2
Headwall Nano-Hyperspec (pushbroom)	Potato yield estimation	400–1000	272	6
ImSpector N17E	Maize kernel vigour assessment	874–1734	–	5
ImSpector V10	Weed identification	400–1000	240	10
OCI-UAV-1000 (pushbroom)	Nutrient assessment in rice	460–983	116	5
ImSpector V10E	Disease monitoring in pears	328–1115	1002	2.8

2.2.7 Performance Metrics and Analyses Used in Later Chapters

No single metric is sufficient for evaluating a disease-detection model. In a controlled benchmark setting, a single summary statistic may be adequate for ranking models. In field biology, however, model choice is almost always multi-criteria: one must consider predictive quality, robustness to hard examples, biological plausibility of the learned features, and runtime or

memory requirements for deployment. This is why the later chapters combine standard summary metrics with cross-validation, class-activation analysis, ablation, and deployment-oriented measures such as GFLOPS, frames per second (FPS), and peak VRAM.

The core quantitative metrics are summarised in Table 2.8. The supporting analyses used to understand *why* a model behaved as it did are summarised in Table 2.9. Read in this way, the empirical chapters are not merely competitions for the highest score; rather, they are attempts to judge whether a model is accurate, well calibrated to the task, likely to generalise, and practical to deploy.

Table 2.8: Plain-language summary of the main quantitative metrics used in later chapters.

Metric	**How it is interpreted in this thesis**	**Later use**
Loss	Measures optimisation error during training and validation. Large train-validation gaps are treated as warning signs of overfitting.	Chs. 3–4
Accuracy	Useful as a coarse summary of correctness, but never treated as sufficient on its own.	Chs. 3–4
Precision, recall, and F1	Used when false positives and false negatives have different practical costs. F1 is the main balanced summary statistic in the later classification studies.	Chs. 3–4
Per-class metrics	Reveal which diseases remain difficult even when the global mean looks strong. This matters because the cocoa diseases differ in visibility and progression.	Chs. 3–4
AUC	Measures separability across thresholds rather than at one decision threshold only.	Chs. 2–4
mAP	Measures localisation quality for detection and segmentation tasks where overlap with the target region matters.	Ch. 2
n parameters/GFLOPS	Used as proxies for model size, overfitting risk, and theoretical compute demand.	Chs. 3–4
FPS/peak VRAM	Used as direct deployment metrics for inference speed and memory demand on realistic hardware.	Ch. 4

Table 2.9: Plain-language summary of the supporting analyses used later in the thesis to interpret model behaviour and compare training strategies.

Analysis	Role in this thesis	Later use
Cross-validation	Repeats training and evaluation across multiple folds so that model quality is judged from a distribution of scores rather than one arbitrary split.	Ch. 3
Confidence intervals, violin plots, and box plots	Show the spread and stability of performance across folds, helping distinguish robust models from fragile ones.	Ch. 3
Grad-CAM maps	Visualise which image regions contributed most strongly to a prediction, allowing inspection of whether a model attends to lesions, pod tissue, or spurious background structure.	Chs. 3–4
Qualitative Grad-CAM scoring	Converts visual inspection of activation maps into a simple 0/0.5/1 score so model attention can be compared systematically across variants.	Ch. 4
Difficult/Unsure relabelling rates	Act as reliable quatitative tests of whether semi-supervised methods generalise beyond the easy core of the training set.	Ch. 4
Ablation studies	Remove or replace architectural or training components one at a time to test which parts of a pipeline are genuinely useful.	Chs. 2–4
Bayesian optimisation and grid search	Used here to systematically explore hyperparameters instead of relying on ad hoc manual tuning.	Chs. 3–4
Random-forest feature importance	Ranks spectral bands by predictive usefulness, helping interpret which wavelength ranges are informative for disease detection.	Ch. 3

2.2.8 Model Testing

2.2.8.1 Black Box Deep Neural Networks

It is well-known how poorly current CV models deal with unexpected edge cases and shifts in test data distribution [193]. However, in applying CV to plant pathology and agriculture, we encounter more cases than most ML practitioners where the test data does not align well with the training data. This is exemplified clearly in chapter 6 where even the models that showed minimal signs of overfitting performed much worse on a truly independent test set than on the

validation set. These problems arise routinely in CV from the effects of camera blur, image quality, or shifting camera angle. However, in plant pathology, we must also contend with the perturbations of weather, climate, plant growth stage, crop variety, a plant's developmental response to growing conditions, and so on. While it remains contentious how robust of a fix techniques such as data augmentations or inductive biases may be to solve the former list of issues [193], the latter issues will only be solved by truly understanding how our models are making predictions.

Although DNNs are still considered black box optimisers, much work has been done to understand their various facets and potential foibles. For example, each dense layer of a CNN has been shown to have distinct roles in feature-level extraction and generalisability [194], and the output of convolution layers have been visualised to show which physical features in an image were more exaggerated [195]. In a similar study, a host of predefined layer-wise and neuron-wise visualisation techniques were applied to a CNN that had been trained on images of plant disease [196]. This work showed that the CNN in question was indeed using visible symptoms of disease that were similar to those used by human experts. Others have sought to learn how best to actively deceive or manipulate a DNN into misclassification. Working within the remit of cybersecurity, it was shown that image classifiers based on SVMs and DNNs could easily be deceived with a simple evasion algorithm [197]. This shows how brittle these classifiers can be and highlights the importance of adopting techniques that rely more heavily on causal inference, such as semi-supervised learning [198] or semantic segmentation. It also highlights the importance of rigorous and conciliatory interrogation of models prior to deployment. At present, our methods of model evaluation are widely considered insufficient, and much more work is needed in this area.

2.2.9 Inspecting Informative Features

A key benefit of the use of CNNs is feature learning. This is the process by which a model will define for itself which features of a data set it considers informative [88]. In other CV algorithms, an engineer must handcraft descriptive features of a subject manually, using their expertise and/or diagnostic tools to guide them. In this latter case, pre-processed data are used rather than raw data, as in a CNN. In the convolution layers of a CNN however,

kernels and attention weights are applied to raw or augmented image data that emphasise informative physical features, and apply inductive biases and self-attention before these data are passed to the dense layer(s) of the network [199]. We might assume that these physical features would include those that humans consider to be the obvious visible markers for plant disease, such as the presence of lesions on a leaf. However, it is likely that these networks will also identify markers that humans do not notice or cannot perceive, and may ignore some features that plant pathologists have long considered important. This provides us with the opportunity to learn more about how to identify disease early with human vision, CV, and molecular biology. Using time-series qPCR, transcriptome or metabolome data to identify the biological markers used by CNNs at the earliest moments of detection would allow for the validation of the image features used by the model. Such a biological explanation of the model's informative features would tell us whether the model was making correct inferences for what we consider to be correct reasons, or whether it was correct for spurious reasons, which would suggest a poor ability to generalise stemming from naive inductive reasoning. Such work may also highlight new ways to identify disease with or without AI or new ways of combating disease spread through phytosanitation, agrochemistry, or plant breeding.

In recent years, the combination of CNNs and transcriptomics in medical research has seen a surge in popularity. Such studies involve spatial transcriptomics [200, 201], the identification of non-small-cell lung cancer subtypes [202], and the elucidation of the various functions of drugs [203]. In plant science, CNNs have been applied alongside transcriptomics in the investigation of gene regulation in *Arabidopsis thaliana L. Heynh* [204]. However, the investigation of the black box nature of CNNs by means of omics appears to be completely absent from the literature.

Attention maps produced by software like Grad-CAM [205, 206] are another way to inspect informative features of image data. Grad-CAM produces an explanation for the decision that a model makes about a given image by visually highlighting the informative features of that image. Grad-CAM is described as "gradient-based" as it uses the gradient data that is fed into the last convolution layer of a CNN. This allows us to make assessments before the spatial relationships in the data are lost in the fully connected layers [205]. Alternative "reference-based" systems, such as DeepLIFT, rely on back-propagation [205] or forward propagation (explanation map) [207], using a reference image that does not contain the

feature of interest. Applying these methods to misclassified images can highlight why a model is performing sub-optimally [196], as results produced with these methods have been shown to be highly correlated with assessments of plant disease made by human experts [207].

2.3 A Roadmap To Commercial Implementation

Once you have developed, trained, and evaluated your model, it is time to begin the process of implementation. However, it is best to have considered and planned this step well ahead of time. There are several decisions made during development that may depend on the intended mode of implementation. For example, if the model is to be run on an edge device or smartphone, computation cost must be kept to a minimum. Likewise, if the model is to be made available via a rented server, reducing computational cost will reduce financial cost. Prior to training, choosing to use architectures such as ResNet18 and MobileNetV3 [208] will help to keep computational cost down and, after training, methods such as pruning and quantisation may reduce this cost further. While Google Colab (Google, Mountain View, California, USA; https://colab.research.google.com/) offers free limited access to GPUs for model training, the rental cost of a 16-GB Nvidia V100 GPU (Nvidia, Santa Clara, California, USA), which would be the minimum needed to train a transformer model or large CNN, is $2.48 USD per hour. As such, developing and training such large models for days, or even weeks, can soon become expensive.

ONNX Runtime (Microsoft Corporation, Redmond, Washington, USA; https://onnxruntime.ai/) offers a huge array of tools to help accelerate, quantise, and deploy trained DNNs. Such models can be incorporated into Android or iOS apps, making use of the phone's built-in camera, and they can be deployed via the web, on edge devices such as a Raspberry Pi, or in embedded systems for drone mapping or smart irrigation. However, the operator schemas supported by ONNX Runtime must be considered here. For example, ConvNeXT, which uses GELUs and stochastic depth, may cause problems as these operators are not yet supported. TensorFlow [209] also offers a pipeline for model deployment, and the PyTorch toolkit for techniques such as quantisation aware training and model compression is maturing but presented difficulties when we attempted to use it. By contrast, the ONNX Runtime pipeline is extremely easy

to use and supports all popular model formats, including PyTorch, TensorFlow, and SciKit Learn [210]. While the latest methods of pruning are reported to achieve a 30% reduction in the size of ResNet18 with only a 2% loss in accuracy on ImageNet [211], this remains an active area of research, producing inconstant results. There is no guarantee that pruning will lessen computational cost. Techniques such as training aware pruning show promise but require further research.

For the implementation of object detection or segmentation models, we recommend the Detectron2 library from Facebook [212]. This library incorporates Faster R-CNN, Mask R-CNN, and some new transformer models such as ViTDet, and offers a host of tutorials on the whole process from training to implementation. In addition, for deploying real-time object detection models, the Ultralytics library [213] provides robust support for the YOLO series (including YOLOv5 to YOLOv11) and transformer-based RT-DETR models. Ultralytics offers a unified application programming interface (API) and comprehensive documentation, facilitating the entire workflow from training to deployment, and supporting a range of tasks including object detection, instance segmentation, pose estimation, and classification.

2.4 Summary

Taken together, this review shows that the state of the art in plant-pathology CV now includes strong off-the-shelf CNN families such as ResNet, EfficientNet, and ConvNeXt, transformer-era vision models, advanced detectors and segmenters such as Faster R-CNN, Mask R-CNN, YOLO, and DETR, and an expanding toolkit for semi-supervised learning, interpretability, deployment, spectroscopy, and molecular validation. These methods have pushed benchmark performance to very high levels on curated datasets, but the literature remains much less mature when the goal is biologically defensible, field-deployable disease detection under realistic constraints.

The most important gaps identified here are therefore not simply about inventing yet another architecture. Rather, the literature still under-serves low-resource deployment, realistic and weakly supervised data acquisition, early disease detection, rigorous validation beyond single summary metrics, integration of biology and spectroscopy with vision models, and careful

interrogation of whether a model is correct for biologically meaningful reasons. These gaps connect directly to the research questions set out in Section 1.10: **RQ1/H1** asks how efficient architectures should be tailored to modest datasets, **RQ2/H2** asks whether non-visible biological signals improve disease discrimination, **RQ3/H3** asks whether advanced training procedures improve hard-case generalisation, and **RQ4/H4** asks whether interpretability and deployment suitability can be improved together.

We describe here all of the tools necessary to develop highly optimised and robust AI models that use minimal computational power and provide real benefit to sectors that have more modest budgets. The application of these tools will allow us to break from the common trend in the AI industry, where expensive hardware is employed to develop complex and computationally expensive models to the detriment of improving training data quality. This review identifies a clear tension in the state of the art: modern AI systems can achieve excellent benchmark performance, yet accuracy on simplified data sets is a poor proxy for robustness, biological validity, and field readiness. The literature shows that the central challenge is not a shortage of powerful architectures, but a shortage of end-to-end workflows that integrate careful data curation, biologically informed sensing, efficient model design, and rigorous evaluation under realistic deployment constraints.

With the application of off-the-shelf architectures to stock data sets, such as the PlantVillage data set [214], we can easily achieve prediction accuracy scores in the high 90% range [215]. However, such models have little value because they will not generalise to complex real-world environments due to the simplicity of the training data and a high likelihood of overfitting. This diagnosis motivates the research questions introduced in Section 1.10. The thesis therefore asks whether lightweight architectures can be tailored more effectively to limited and noisy plant-pathology data, whether non-visible sensing contributes complementary evidence for disease discrimination, whether advanced training procedures improve hard-case generalisation, and whether interpretability and deployment suitability can be improved together rather than treated as separate concerns.

We offer the following recommendations for the development of efficient, inexpensive, and robust CV models for plant pathology. These recommendations are framed as practical design principles for the preliminary investigations and contribution chapters, rather than

as abstract methodological advice.

Garbage in, garbage out: The thoughtless application of advanced models to poorly labelled, simplistic, contaminated, or inappropriately transformed data will yield models that have little value in the field, with slow inference times, poor accuracy, and an inability to generalise. To avoid this fate, we should: (A) where possible, consult with specialists and utilise the invaluable tools from biology, chemistry, and spectroscopy to label data; (B) use the minimum appropriate image input size to improve runtime speed and help avoid overfitting; and (C) avoid needless data transformations such as normalisation, which can alter data in unreliable ways.

The potential in training procedures: Techniques such as semantic segmentation and semi-supervised learning have the potential to lessen both bias and variance in a model's predictions by promoting deductive reasoning over inductive reasoning. While appropriately scaled CNNs and evolved neural networks offer the potential to produce models with optimised runtime speed and improved generalisation ability.

Robust and conciliatory interrogation of models: While simpler modelling methods, such as SVMs, still have a role to play in modern CV, most of the models we employ for this purpose are exceedingly complicated and are prone to failing in equally complicated ways. Failure of a disease detection model resulting in an outbreak of disease could have very serious consequences. It is therefore vital that we rigorously test the models we develop to ensure that they are not prone to misclassification born of overfitting and naive generalisations. While metrics such as accuracy, F1, area under the receiver operating curve (AUC), recall, and precision are valuable, DNNs are often capable of learning to optimise these summary statistics indirectly, rather than learning to produce reliable predictions. Tools such as confusion matrices, cross-validation, and explanation maps go much further in understanding the behaviour of CV models. However, it is important that we invest in the development of new and tailored means of understanding these models, such as the application of omics, as discussed in section 2.2.9.

The next chapter, Chapter 3, takes these gaps forward through a series of preliminary investigations into semantic segmentation, outlier filtering, and normalisation-related design choices. Those exploratory studies establish the methodological direction for the main con-

tribution chapters, beginning with non-visible sensing in Chapter 4, then architecture in Chapter 5, and finally training in Chapter 6, in which the research questions are addressed directly.

If we apply our wealth of knowledge and proven techniques from botany and agronomy to the acquisition of training data, the development of data-processing pipelines, and the interrogation of trained models, we can produce applications with game-changing potential. We are now only 27 years away from a predicted global population of 9.7 billion people [216]. Thus, with the devastating effects of the climate crisis already very much apparent, it is vital that we act now to build robust international infrastructure targeted at securing food supplies and eliminating extreme poverty. The techniques discussed here may enable us, as a community of growers, botanists, and AI developers, to help reduce poverty, improve the relationship between growers and the natural environment, and increase stability in the agriculture industry from the foundation up.

Chapter 3

Preliminary Investigations

"Truth emerges more readily from error than from confusion."

Francis Bacon

3.1 Introduction

The literature review in Chapter 2 identifies several questions that were important to test before committing to the later contribution chapters. The studies collected here were exploratory rather than definitive: they were designed to establish whether particular directions were promising enough to inform the later thesis narrative or whether they should instead motivate future work. In that sense, this chapter provides the bridge between the review of prior work and the later spectroscopy, PhytNet, and advanced-training chapters.

The four preliminary investigations reported here examine semantic segmentation for cocoa disease detection, outlier filtering for noisy web-scraped plant imagery, the effects of image and batch normalisation on disease classification, and the optimisation of batch-normalisation momentum and image input size. Together, these studies informed the later spectroscopy, architecture, and training chapters in Chapters 4 to 6 and helped clarify which ideas were sufficiently mature for direct contribution and which remained exploratory.

3.2 Preliminary Investigation 1: Semantic Segmentation for Cocoa Disease Detection

Motivated by the object-detection and segmentation literature reviewed in Section 2.1, we applied Mask region-based convolutional neural network (Mask R-CNN) to the task of segmenting images of diseased cocoa trees. The training dataset consisted of 186 images of black pod rot (BPR), 121 images of frosty pod rot (FPR) and 63 images of witches' broom disease (WBD). The model was trained, starting with the "mask rcnn R 50 FPN 3x" weights, for 1,000 epochs.

The preliminary results from this investigation were somewhat encouraging. However, although the selected positive results in fig. 3.1 show that this model has the potential to perform well, these results are not representative of the full testing set. The average precision per class was 4.29, 13.45 and 30 for BPR, FPR and WBD respectively. *i.e.* the model performed acceptably on WBD, despite the low number of training images, but poorly on most cases of BPR and FPR.

Notwithstanding the potential theoretical benefits discussed in Chapter 2, manual annotation of a full training dataset with masks is extremely laborious. So without the promise of improved results, relative to a simple convolutional neural network (CNN), this additional effort may not pay. However, considering the favourable preliminary results in this study and one other [133], with the incorporation of automated annotation tools and/or semi-supervised learning, semantic segmentation shows promise as an avenue of research for computer vision (CV) in plant pathology.

(a) Black pod rot (b) Frosty pod rot (c) Witches' broom

Figure 3.1: Application of semantic segmentation with Mask R-CNN to highlight whole trees infected with (a) black pod rot, (b) frosty pod rot and (c) witches broom disease. The percentage scores show the degree of confidence in the model's diagnosis.

3.3 Preliminary Investigation 2: Outlier Detection and Semi-Supervised Filtering

3.3.1 Generative Filtering with variational autoencoders (VAEs)

The review in Section 2.1.8 highlighted both the appeal and the limitations of reconstruction-based filtering for contaminated image collections. To test this more directly, we trained Nouveau variational autoencoder (NVAE) on the curated forestry and arable images from Google and Bing (FAIGB) plant images and found that it could reconstruct any image in the ImageNet dataset with binary cross-entropy loss similar to that of genuine plant images. This supports the concern that reconstruction error alone is not a reliable indicator of whether an image is biologically relevant.

As an alternative to NVAE, we attempted to use a custom convolutional VAE with a ResNet152 [6] backbone to apply the two outlier-detection methods discussed in Section 2.1.8. However, we were unable to get this architecture to function well enough to sufficiently compress the data and reconstruct images with high fidelity.

3.3.2 Semi-Supervised Binary Filtering

As an alternative to using a variational autoencoder for outlier detection, we trained a semi-supervised binary Outlier-NoneOutlier (in this case, "plant" or "non-plant") classifier, which achieved near-perfect results. We used the ResNet18 architecture and initially trained it on the curated FAIGB dataset of 31,225 plant images and an equal-sized random subset of the ImageNet dataset, which constituted the non-plant images. We then continued training using the below algorithm and a larger contaminated crawl of 96,692 candidate images generated by the same scraping workflow.

while $nRelabledImages > 0$ **do**
  train model
  for image in ContaminatedImages **do**
    classify image
    if $ClassificationConfidence \geq 99\%$ **then**
      label image
      add image to training set
    end if
  end for
end while

During this process, 1,376 non-plant images and 44,212 plant images from the contaminated dataset were correctly labeled by the model. After the first round of semi-supervised training completed, images that this model classified with >99% confidence were manually reviewed. Incorrectly labeled images were manually re-labeled and a second round of semi-supervised training was begun. After the first round of semi-supervised training, classification of images as "plant" with >99% confidence was >99% accurate but classification of images as "non-plant" with >99% confidence was only about 50% accurate.

After the second round of semi-supervised training, the model performed with >99% accuracy and F1 score (F1) score for both classes. Thus showing a clear superiority in this technique's ability to identify contaminant images over the VAE approaches. This is in addition to its ease of implementation, and reduced training time and compute requirements. After training, the model was used to classify all 96,692 images in the contaminated dataset.

In practice, this filtering workflow formed part of the broader process by which noisy web-scraped imagery was refined into a cleaner plant dataset for subsequent experiments.

3.4 Preliminary Investigation 3: Normalisation Choices for Disease Detection

To test the concerns about image and batch normalisation raised in Section 2.1.14, we conducted an ablation study with ResNet18 and ConvNeXt_tiny (table 3.1) to assess the effects of image normalization (IN), batch normalization (BN) and layer normalization (LN) in disease detection. BN in ResNet18 increased training speed by 2.39 times, while IN slowed training by 1.74 times. IN did not affect training time in ConvNeXt_tiny. We also found that BN improved stability in training, as assessed by plots of training and validation loss. However, IN decreased the F1 score by 0.76% and 0.34% in ConvNeXt and ResNet18 respectively, and increased overfitting. Removal of BN in ResNet18 decreased F1 by 1.92% but the ConvNeXt model, in which BN is replaced with LN, had an F1 score 2.84% higher than ResNet18 with BN. Therefore simply deactivating the BN layers in ResNet18 led to worse results in every metric. However, the use of LN instead of BN in ConvNeXt appears to have had no deleterious effect. The removal of the IN transformation, which occurs prior to data input, improved the performance of both model architectures for disease detection in all metrics, including training time and overfitting.

These results are unsurprising if we consider the effect of image normalisation shown in fig. 3.2. Here, IN distorts the colour of the cocoa pods and obscures much of the large lesions that are clearly visible in the original images. This effect may not prevent a CNN from identifying these objects as cocoa pods or trees by their shape but it does obscure many subtle disease symptoms that are necessary for the detection of early disease development. The above ablation study was conducted with a dataset of late-stage disease, from which the fig. 3.2 images were sampled. So if early disease detection were required, the differences between these methods may have been more pronounced. Additionally, BN has been observed to introduce unacceptable levels of error when batch size is small [111]. This is an important issue to consider for generative models, CV models with video or high-resolution images or

when compute resources are limited.

Table 3.1: Results of an ablation study to assess the effects of image normalisation and batch normalisation on disease detection performance.

Image Norm	Batch Norm	Layer Norm	Train time (m)	Loss (%)	Acc (%)	Recall (%)	Precision (%)	F1 (%)
ConvNeXt Tiny								
No	–	Yes	1344	0.290	88.25	88.25	88.82	88.14
Yes	–	Yes	1368	0.322	84.51	87.51	88.14	87.38
ResNet18								
No	Yes	–	739	0.361	85.41	85.41	86.17	85.30
Yes	Yes	–	1088	0.380	85.18	85.68	84.96	84.96
No	No	–	1764	0.412	83.49	83.49	84.05	83.38

(a) (b)

Figure 3.2: (a) Original and (b) normalised images of cocoa pods showing various stages of disease development. Note the effect of normalisation on one's ability to see disease symptoms.
Normalisation of pixel values was carried out with the following means are variance values: mean: (0.485, 0.456, 0.406) variance (0.229, 0.224, 0.225)

3.5 Preliminary Investigation 4: Batch-Normalisation Momentum and Image Size

We also ran a hyperparameter optimisation sweep using the Weights and Biases platform (WANDB) [217], which included the BN momentum hyperparameter and image input size (fig. 3.3). The model architecture used was ResNet18 [6] and the dataset included the following four classes: black pod rot, frosty pod rot, healthy cocoa and witches broom

disease with a 90:10 split and training set size of n = 271, 266, 436 and 92 respectively. One hundred models were trained with these hyperparameters randomly sampled from predefined ranges (Image size: 124:1224 pixels, BN mom.: 0, 10^{-5}:0.9). We also used WANDB to run a random forest regression with the validation F1 as the dependent variable and the two hyperparameters as independent variables. From this, an importance score was calculated for each hyperparameter on a scale of 0-1. The highest performing model scored validation F1:0.75 and area under the curve (AUC):0.87. Additionally, the per-class F1 score for healthy cocoa was 0.88, showing a strong ability to detect non-specific disease.

While the importance of image size (0.694) is not surprising, the BN momentum score (0.306) is quite low. This casts doubt on the assertion that optimisation of BN momentum can have much impact in lessening the deleterious effects of BN. However, this result and that of the optimised BN momentum value (0.001) (fig. 3.3A), suggests that this hyperparameter should be optimised, rather than relying on the default value of 0.1. Training the same model with a BN momentum set at 0.1 yielded an F1 score of 0.737, *i.e.* a 1.3% decrease relative to the optimised value.

This study also provides an optimised image input size for mid- to late-stage disease detection, using ResNet18, of 277 pixels^2 (fig. 3.3B), though this should be optimised for each use case. Previously, image compression has been said to have minor effects on disease detection [218], while elsewhere it is suggested that image compression should even be avoided completely for small symptoms [219] or kept above an arbitrary 1 megapixels (1,000 x 1,000 pixels) [220]. However, with the present dataset, which contains images of diseases at varying degrees of progression, using an image size greater than 277x277 was deleterious to validation F1 score. This is in addition to the reduced image size providing faster runtime in training and inference and a reduction in overfitting.

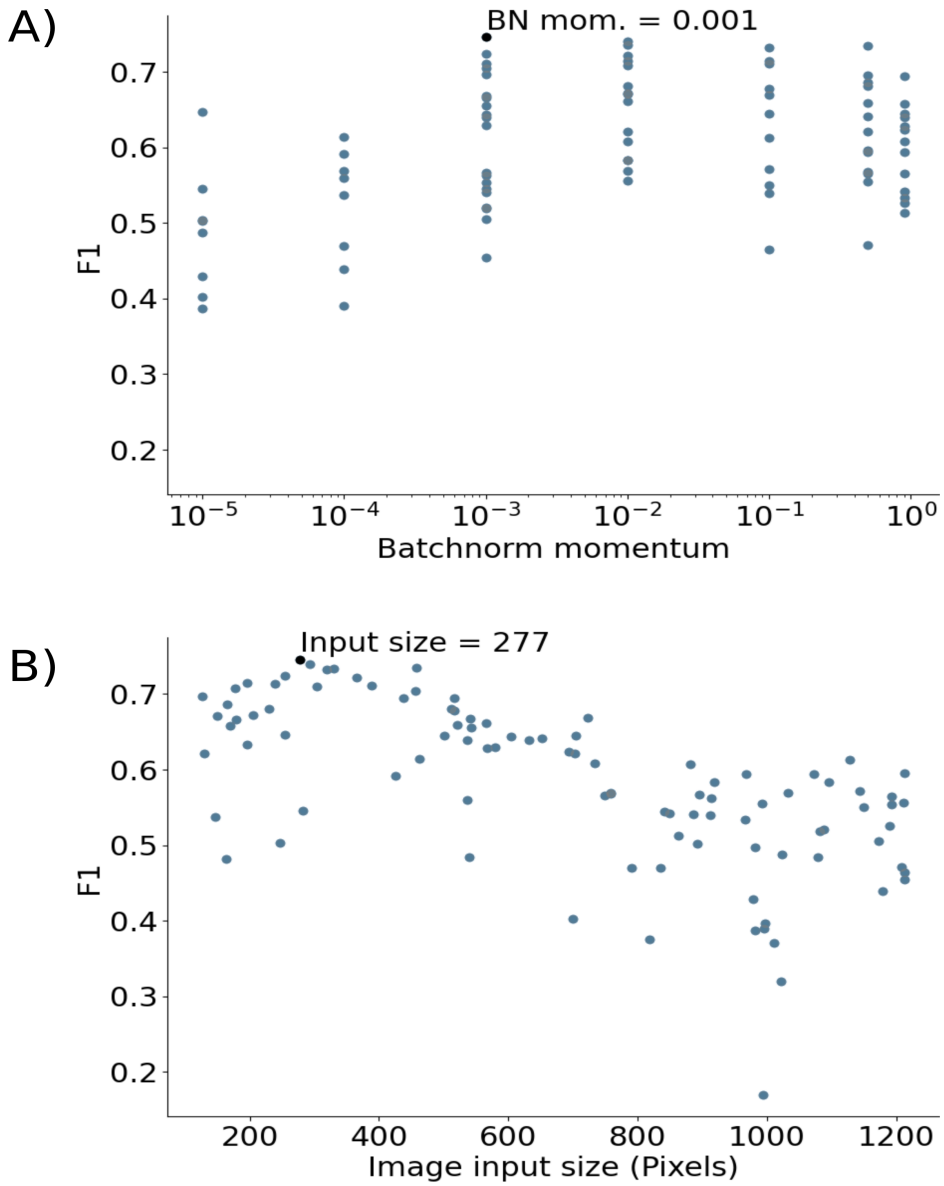


Figure 3.3: Results of a hyperparameter optimisation sweep training 100 ResNet18 models for disease detection in cocoa trees with variable batchnorm momentum (A) and square image input size (B). The optimisation sweep randomly sampled from distributions of the two variables concurrently. Beginning with the ImageNet1KV2 weights, the models were trained on a dataset of 1,065 images of the following four classes. Black pod rot [271], Frosty pod rot [266], Witches broom disease [92] and Healthy cocoa [436]. The optimised validation F1 score was 0.75.

Chapter 4

Spectroscopy and Non-Visible Signals for Cocoa Disease Detection

"Measure what is measurable, and make measurable what is not so."

Galileo Galilei

4.1 Introduction

This chapter addresses **RQ2/H2** from Section 1.10: whether infrared and spectroscopy-derived signatures provide useful complementary information for cocoa disease analysis beyond standard visible-spectrum imaging. However, it is not a dedicated early-detection chapter. Instead, it reports a series of exploratory experiments on healthy tissue and on cocoa tissues showing readily apparent, approximately mid-stage disease symptoms, in order to test whether different parts of the tree or different wavelength regions contained informative signal at all. In cocoa pathology, visible symptoms can be subtle, heterogeneous, and stage-dependent. Before attempting a robust early-detection study, however, it was first necessary to establish whether non-visible measurements remained informative even when disease was already diagnostically evident.

This chapter builds on the spectroscopy background reviewed in Section 2.2.6 and on the exploratory design choices summarised in Chapter 3. It appears before the PhytNet architecture chapter because these non-visible sensing experiments helped frame the later infrared-image comparisons in Chapter 5. Rather than assuming that infrared or spectroscopy necessarily provide useful complementary signal, this thesis first tests whether such signal is detectable and stable enough to justify stronger downstream modelling claims. In practical terms, that means asking whether disease can be inferred from photosynthetic measurements taken away from clearly symptomatic tissue and whether pod reflectance spectra from readily apparent disease states contain wavelength bands that are informative for discrimination.

This chapter therefore proceeds in two stages. First, MultispeQ measurements are used to test whether diseased trees show detectable changes in photosynthetic behaviour away from the visibly affected tissue. Second, field spectroscopy is used to test whether pod reflectance spectra contain a disease signal that remains informative once leakage between collection batches is controlled. The discussion then interprets what these results imply for the feasibility of later early-detection work, for the infrared-image experiments in Chapter 5, and for the limits of the present spectroscopy dataset.

4.2 Methods

4.2.1 Spectroscopy

We used a MultispeQ v2.0 [181] to measure photosystem II quantum yield (Phi2) and non-photochemical quenching (NPQt) in cocoa trees with different disease states. Both Phi2 and NPQt have been shown to have significant negative and positive correlations with disease index respectively [181]. These measurements were taken to assess if non-visible signals of disease could be detected in the foliage of cocoa trees. Readings were taken from 1) healthy leaves on trees with no signs of disease nearby (n=9), 2) leaves clearly infected with witches' broom disease (WBD) (n=5) and 3) the nearest leaf to pods or leaf tissue infected with black pod rot (BPR), frosty pod rot (FPR) or WBD (n=10, 5, and 5 respectively). All measurements in this chapter were collected during a single field day across multiple farms in the same wider growing area northeast to southeast of Guayaquil, Ecuador. The design

was therefore cross-sectional rather than longitudinal: no early/late disease time points were defined, and the diseased tissues sampled here were selected because symptoms were already readily apparent.

To measure the reflectance spectrum of diseased and healthy cocoa pods, we used a field spectrometer, produced by tec5 (Steinbach, Germany). The processed spectra used here spanned 256 wavelength bands from 302–1146 nm. The spectroscopy data was gathered during the same single field day on four farms located east of Guayaquil, Ecuador, near the foothills of the Andes mountains. Readings were taken following the manufacturer's instructions. This was not a longitudinal study: the three acquisition batches reflect collection groupings within that one field campaign rather than biologically distinct time points, and no early-, mid-, or late-stage labels were defined beyond the practical selection of pods whose disease symptoms were already clearly visible. Each spectrum was stored as a CSV file within a batch-specific directory tree. During ingestion, only files containing the columns Wvl_Obj, Counts_Obj, and Counts_Ref were retained, spectra with non-numeric values, non-positive reference counts, or mismatched wavelength grids were excluded, and reflectance was computed as Counts_Obj/Counts_Ref.

The retained dataset contained 111 pod spectra from three collection batches: healthy cocoa pods (Healthy, n=47), pods with black pod rot symptoms (BPR, n=40), and pods with frosty pod rot symptoms (FPR, n=24). To avoid leakage between spectra collected in the same batch, model evaluation used leave-one-batch-out cross-validation, with batch identity as the grouping variable. In each fold, a scikit-learn random forest classifier [210] was trained with 61 trees, maximum depth 10, minimum samples per split 8, minimum samples per leaf 2, class weight set to balanced_subsample, and all available CPU cores enabled; all other parameters were left at library defaults. The forest random state was set to 101, 102, and 103 across the three folds. Performance was reported as fold-wise and overall accuracy, balanced accuracy, macro- and weighted-F1 score (F1), precision, recall, Cohen's kappa, and one-vs-rest ROC-AUC where out-of-fold probabilities were available. Uncertainty in the overall metrics was estimated by bootstrap resampling of the out-of-fold predictions (5,000 resamples).

To assess whether any wavelengths were stably informative, permutation importance was

computed on each held-out batch using balanced accuracy as the scoring function, with 40 repeats per fold. Mean importance curves were summarised with 95% bootstrap confidence intervals across folds. The mean reflectance spectra were plotted as batch-level means with 95% confidence intervals estimated across batches. For additional interpretation, the grouped-validation analysis was also repeated 120 times for each of five spectral settings: ultraviolet (UV) (302–399 nm), visible (400–699 nm), NIR (700–999 nm), the upper 1000–1146 nm interval, and the full spectrum. These repeated grouped runs were compared using Mann-Whitney U tests, Kolmogorov-Smirnov tests, Vargha-Delaney A effect sizes, and Holm correction for multiple comparisons.

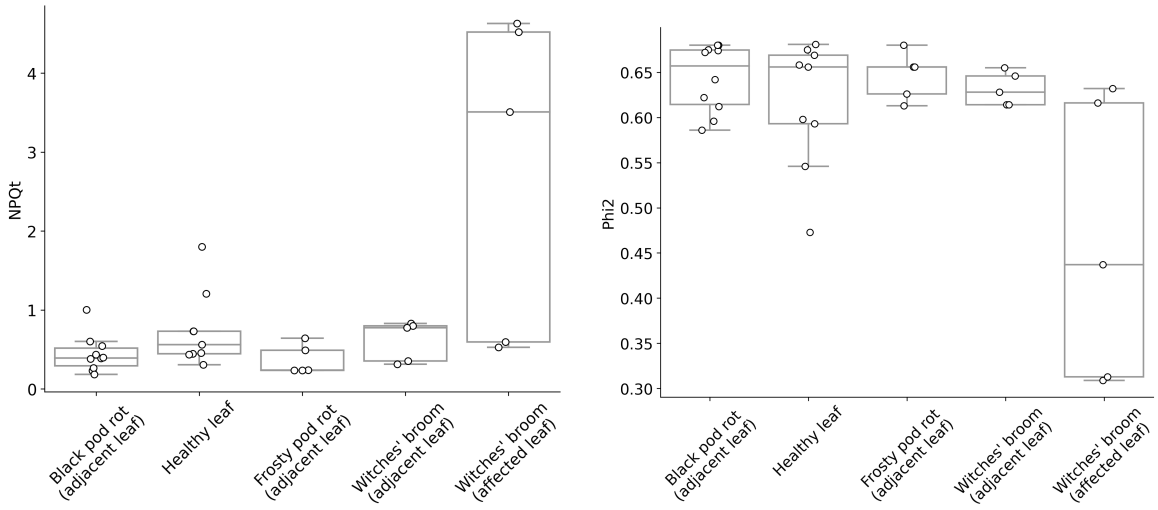
4.3 Results

4.3.1 Locating Machine Visible Symptoms

Figure 4.1 shows the result of measuring Phi2 and NPQt in diseased and healthy cocoa trees using a MultispeQ V2. These measurements were taken to assess if non-visible signals of disease could be detected in the foliage of cocoa trees. While some of the leaves that were visibly infected with WBD showed clear but inconsistent signals of compromised photosynthesis, the leaves adjacent to pods and/or leaves infected with BPR, FPR or WBD showed no signs of reduced Phi2 or increased NPQt, relative to healthy trees. As such we see no evidence here that these diseases can be detected through compromised photosynthesis in foliage that shows no human visible signs of disease, even when disease elsewhere on the tree is already readily apparent.

4.3.2 Grouped Evaluation of Pod Reflectance Spectra

Figure 4.2 shows class mean reflectance spectra with 95% confidence intervals estimated between collection batches. The mean curves suggest some separation, especially a reduction in longer-wavelength reflectance for frosty pod rot pods, but the intervals are wide and overlap substantially. This indicates that any disease-related spectral differences were not stable across the three collection batches.



(a) Non-photochemical quenching

(b) Quantum yield of Photosystem II

Figure 4.1: Distributions of non-photochemical quenching (NPQt)(a) and photosynthetic yield (Phi2)(b) of cocoa trees with different disease states. Box plots show the interquartile range with whiskers at 1.5 times the IQR from the first and third quartiles. Raw data points are plotted as white circles. Measurements taken from cocoa trees in five disease states with a MultispeQ v2.0. BPR n=10, FPR n=5, Healthy n=9, Witches' broom (adjacent tissue) n=5, Witches' broom (affected tissue) n=5.

More importantly, leave-one-batch-out validation showed weak cross-batch generalisation. Fold accuracies were 0.349, 0.360, and 0.278, giving a mean accuracy of 0.329 ± 0.045. The corresponding mean balanced accuracy and macro-F1 were 0.377 ± 0.011 and 0.281 ± 0.049 respectively. Across all out-of-fold predictions, overall accuracy was 0.342, balanced accuracy 0.317, macro-F1 0.316, weighted-F1 0.340, and one-vs-rest macro/micro ROC-AUC 0.469 and 0.497. A majority-class baseline that always predicts Healthy achieves 0.423 accuracy, so the random forest does not outperform that simple reference on raw accuracy.

Bootstrap 95% confidence intervals were correspondingly broad: 0.252–0.432 for accuracy, 0.232–0.404 for balanced accuracy, 0.228–0.404 for macro-F1, 0.252–0.428 for weighted-F1, and 0.389–0.550 for macro ROC-AUC. These intervals span near-chance behaviour and reinforce that the observed performance is not robust.

Figure 4.3 shows that FPR recall was particularly poor, with only 4 of 24 samples classified correctly (recall 0.167, F1 0.195), and that healthy pods were often misclassified as black pod rot. This confusion pattern suggests that the model was not capturing disease-specific structure that transferred reliably across collection batches.

Permutation importance was also weak. As shown in Figure 4.4, the largest mean-importance peaks occurred around 382.94, 403.00, and 943.19 nm, but their lower confidence bounds still touched zero. Only one wavelength, 754.72 nm, had a confidence interval whose lower bound remained above zero, and its effect size was still small. Band-level importance summaries ranked UV highest (mean importance 0.00470), followed by visible light (0.00355), NIR (0.00065), and the 1000–1146 nm interval (-0.00010), again indicating that no single long-wave region dominated the result.

Across 120 repeated grouped runs, UV produced the highest mean balanced accuracy (0.3580), visible and the 1000–1146 nm interval were very similar (0.3330 and 0.3319), the full spectrum was lower (0.3135), and NIR was lowest (0.2773). Holm-adjusted pairwise tests showed that visible and the 1000–1146 nm interval were statistically indistinguishable on balanced accuracy, whereas both clearly outperformed NIR. UV outperformed all other spectral settings. These repeated-run comparisons therefore reinforce that the strongest signal in this dataset lies in the shorter wavelengths, not in NIR or the measured SWIR-edge interval.

4.4 Discussion

4.4.1 Photosynthetic Activity as a Disease Indicator

We observed no significant changes in photosynthetic yield or non-photochemical quenching between healthy leaves and those adjacent to diseased pods or leaves. This suggests that, if there exist systemic effects of these diseases on photosynthesis, they were not readily detectable under the cross-sectional, mid-stage conditions examined here. This result reinforces the need to focus on localised symptoms for disease detection in cocoa, which is consistent with what we know of the pathology of these diseases, *i.e.* while *Phytophthora spp.* can cause seedling blight and trunk cankers in adult trees, BPR and FPR symptoms tend to be isolated to the infected pods [221, 222]. However, the sample sizes here were small and the apparent differences were inconsistent, so the safest interpretation is simply that no reliable foliar photosynthetic signal was demonstrated in this experiment.

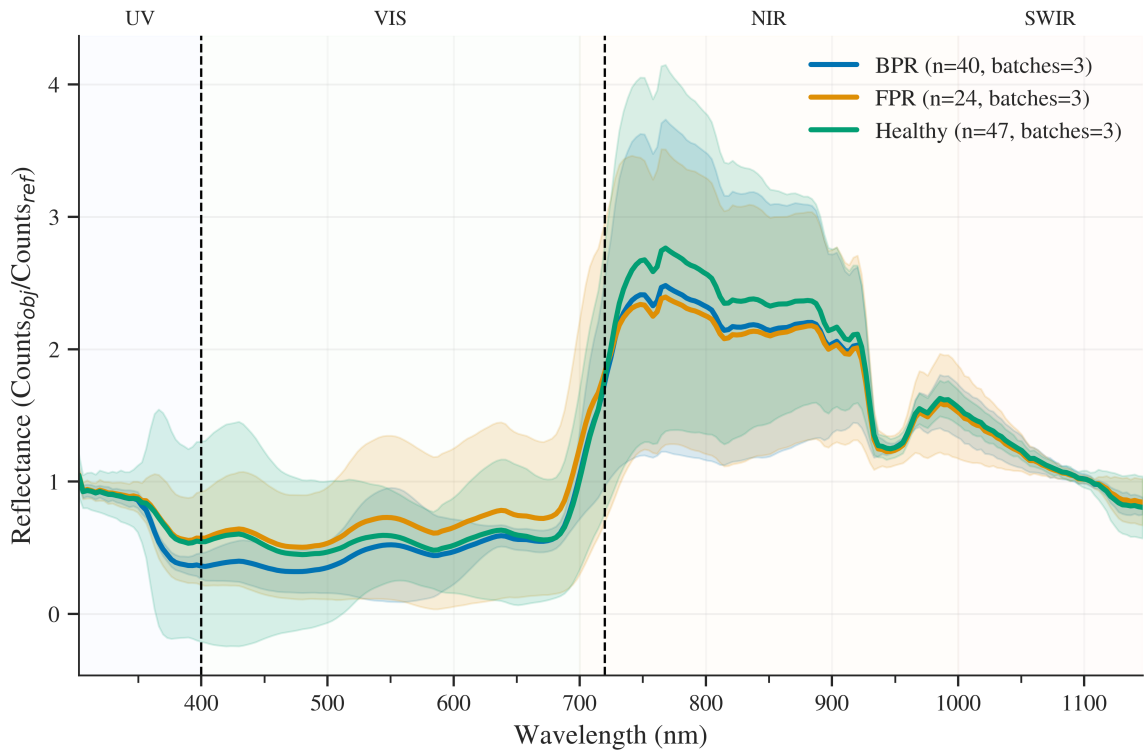


Figure 4.2: Class mean reflectance spectra for healthy, black pod rot, and frosty pod rot cocoa pods. Curves show batch-level means and shaded regions show 95% confidence intervals estimated across the three collection batches. Background shading marks the UV, visible, NIR, and upper 1000–1146 nm wavelength intervals used in the grouped analysis.

4.4.2 Spectral Characteristics of Cocoa Pod Diseases

Under leakage-safe grouped validation, the reflectance spectra did not provide robust evidence of reliable disease classification. The apparent reduction in longer-wavelength reflectance for frosty pod rot remained visually suggestive, but it was not sufficiently stable across batches to support a strong biological interpretation. The low balanced accuracy, weak FPR recall, and substantial confusion between healthy and BPR spectra indicate that batch-to-batch variation currently rivals or exceeds disease-related variation in this dataset.

These results also temper any claim that NIR or short-wave infrared wavelengths were clearly more informative than shorter wavelengths. Although one isolated NIR-adjacent wavelength at 754.72 nm remained marginally positive under the confidence-interval filter, the broader NIR band still performed worst in the repeated grouped analysis, and the 1000–1146 nm interval was no better than the visible band. UV produced the strongest and most repeatable

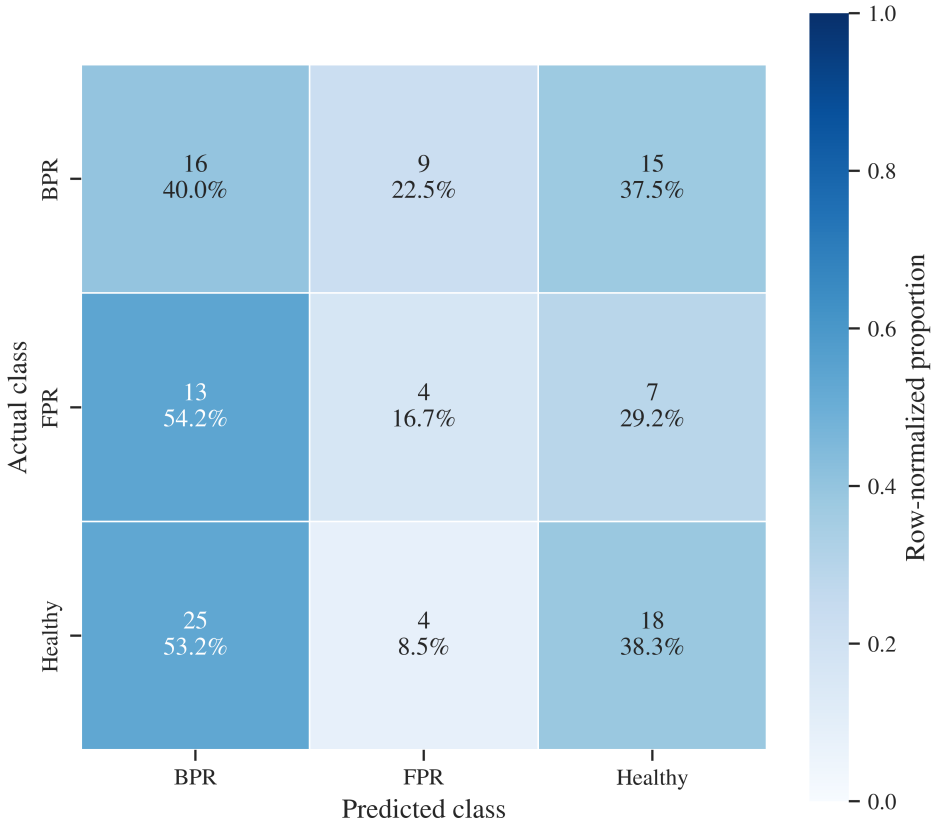


Figure 4.3: Out-of-fold confusion matrix from leave-one-batch-out cross-validation. Entries show counts and row-normalised percentages for the aggregated held-out predictions across the three batches.

performance. On the present data, there is therefore no compelling evidence that NIR or the measured SWIR-edge interval is the dominant source of disease information.

This does not imply that non-visible sensing is useless for cocoa disease detection. Rather, it shows that the present spectroscopy dataset is too limited and batch-sensitive to support strong general conclusions, even when symptoms are already readily apparent. That is precisely why this chapter remains exploratory and why it does not attempt to make a strong early-detection claim: if robust differences are difficult to demonstrate at approximately mid-stage disease, then focusing this part of the thesis on earlier symptoms would have been premature. The later infrared-image comparison in Chapter 5 may still detect a narrower and more practically useful non-visible cue, especially for FPR, but the spectroscopy evidence here should be treated as exploratory and hypothesis-generating rather than confirmatory.

The immediate implication for future work is methodological. More independent collection

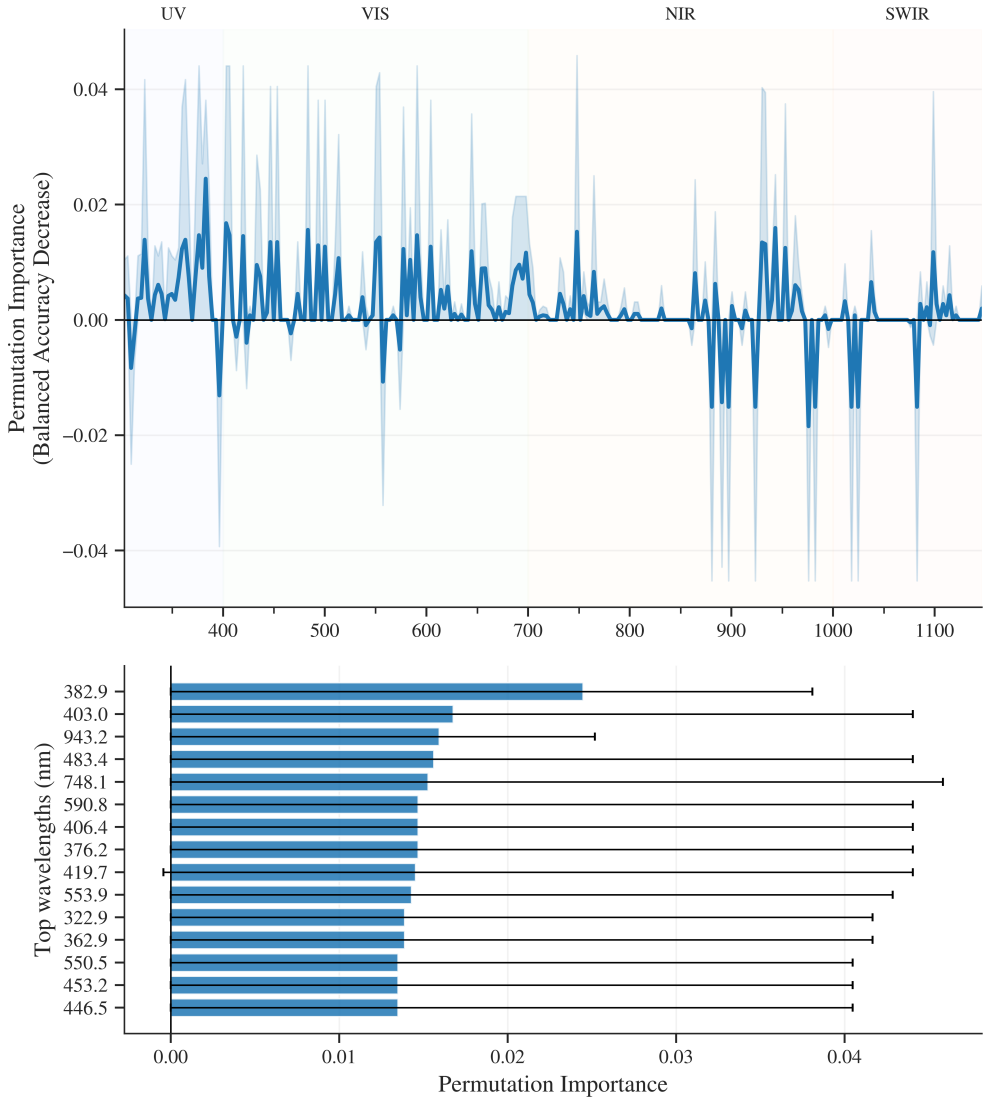


Figure 4.4: Permutation importance of wavelength bands under leave-one-batch-out evaluation, measured as the decrease in balanced accuracy when each wavelength was permuted within the held-out batch. The upper panel shows mean importance with 95% bootstrap confidence intervals across folds, and the lower panel shows the top 15 wavelengths. The largest peaks by mean importance were not robust; only one wavelength, 754.72 nm, had a confidence interval whose lower bound remained above zero, and its effect size was small.

batches are needed, not merely more spectra from the same few batches. In addition, batch harmonisation or domain-adaptation methods, together with simpler or more strongly regularised classifiers, should be tested before wavelength-importance patterns are interpreted biologically. Better control of cultivar, pod temperature, disease stage, and ambient conditions would also help disentangle genuine disease signal from acquisition variability.

4.5 Summary

Taken together, these experiments show that not all non-visible measurements are equally informative for cocoa disease detection, and that the spectroscopy evidence presented here is not conclusive. This chapter was designed as an exploratory study of healthy tissue and readily apparent, approximately mid-stage disease rather than as a longitudinal early-detection study. MultispeQ readings from adjacent foliage did not reveal a reliable disease signal. Leave-one-batch-out evaluation of pod reflectance spectra likewise produced weak cross-batch generalisation and only minimal robust wavelength importance, so **RQ2/H2** is not strongly supported by this dataset alone. The chapter therefore narrows, rather than resolves, the non-visible sensing question and provides the cautionary context for the later infrared-image experiments: infrared imagery may still capture useful information, but stronger claims about spectroscopy will require more independent batches, tighter experimental control, and batch-robust modelling.

Chapter 5

Tailoring Convolutional Neural Networks

"Grown-ups love figures. When you talk to them about a new friend, they never ask questions about essential matters. They never say to you: 'What does his voice sound like? What games does he prefer? Does he collect butterflies?' They ask you: 'How old is he? How many brothers does he have? How much does he weigh? How much does his father earn? It is only then that they feel they know him."

Antone de Saint Exupéry, The little Prince

5.1 Introduction

5.1.1 Background

This chapter follows the non-visible sensing analysis in Chapter 4 and provides the most direct test of **RQ1/H1** in Section 1.10: whether a tailored lightweight convolutional architecture can match or exceed the practical performance of larger off-the-shelf models while reducing computational cost. That question matters because cocoa disease detection is not a setting in which one can assume abundant labels, controlled imaging conditions, or access to expensive hardware. In this setting, choosing the newest or largest model by default is not a principled strategy; model capacity has to be matched carefully to the size, noise level, and biological realism of the data.

This chapter builds directly on the architecture, optimisation, and evaluation background reviewed in Chapter 2, on the exploratory findings gathered in Chapter 3, and on the cautionary non-visible sensing results presented in Chapter 4. Those earlier chapters narrowed the design space by showing which ideas looked promising enough to justify deeper investment and by showing that broader spectroscopy evidence was not yet strong enough to carry the main argument about non-visible sensing on its own. Chapter 5 therefore takes the next step by committing to an architectural solution, developing PhytNet, and evaluating whether a carefully tailored convolutional neural network (CNN) can offer a better balance of accuracy, robustness, and runtime suitability than better-known generic competitors.

5.1.2 Fitting Architectures to Datasets

As discussed in Section 2.1.11, projects with small training datasets require the size and complexity of the neural network to be matched carefully to the size and signal quality of the dataset. Otherwise, a model may memorise background structure, noise, and outliers rather than learning genuine disease features, yielding deceptively strong training performance but poor real-world generalisation. This chapter addresses that practical problem by comparing PhytNet against representative architectures already motivated in Chapter 2: ResNet18, EfficientNet, and ConvNeXt [6][163] [94]. This chapter is centred on **RQ1/H1**,

with a supporting link to **RQ4/H4** through its gradient-weighted class activation mapping (Grad-CAM)-based analysis of model behaviour. The broader non-visible sensing question posed by **RQ2/H2** has already been examined in Chapter 4, so the infrared comparison reported here should be read as a narrower image-based follow-up rather than as the main test of that hypothesis.

The chapter proceeds in a deliberate sequence. It first introduces the image datasets that define the modelling problem, then describes the development of PhytNet and the optimisation workflow used to tailor it to the data, and then compares PhytNet with four competing models using cross-validation, loss, F1 score (F1), computational cost, and Grad-CAM analysis. The discussion then interprets what those results mean for low-resource cocoa disease detection, for the more cautious non-visible sensing picture established in the previous chapter, and for the broader thesis argument about matching architectures to realistic data rather than to benchmark convenience.

The reader should therefore expect more than a simple model leaderboard. The central claim tested here is that, on a small and noisy cocoa disease dataset, a well-tailored lightweight architecture may be more valuable than a nominally stronger but poorly matched alternative. The infrared-versus-red, green, and blue (RGB) comparison is included as a narrower follow-up to the preceding non-visible sensing chapter, not as standalone proof that non-visible sensing is robustly superior. Questions about advanced training procedures, including semi-supervised relabelling and harder-case reweighting, are deferred to Chapter 6; here the emphasis is on establishing the architectural foundation on which those later training choices can be judged.

5.2 Methods

5.2.1 Image Data Collection

We collected two datasets of infrared (IR) (n=240) and RGB (n=240) images taken concurrently with the same Olympus OMD EM-5 II camera. While the IR data was collected primarily for model development, the additional RGB data allowed us to compare an IR- and

RGB-trained ResNet18 model for disease classification. Images were of healthy cocoa trees bearing pods as well as trees bearing pods with black pod rot (BPR), frosty pod rot (FPR) and witches' broom disease (WBD) in equal numbers. Diseased pods showed clearly visible and easily diagnosable symptoms in the early to mid stages of disease development. We randomised across a variety of factors such as geographic location, disease stage, crop variety and air temperature. The two datasets were collected with and without a 720 nm neutral density filter to block all visible light. For the RGB images, the factory camera settings were used with the "intelligent-auto" program selected. A tripod was used to collect the IR images to allow for longer exposure and the following camera settings were applied. ISO: 200, white balance: 2000 kelvin, noise reduction: On, noise filter: low, shutter speed: auto, delay: 12 seconds. These images were collected at two research stations, on either side of the Andes. The research stations, belonging to Instituto Nacional de Investigaciones Agropecuarias (Instituto Nacional de Investigaciones Agropecuarias (INIAP)), were located at Pichilingue and Coca, Ecuador. Using the INIAP stations at Pichilingue and Coca introduced geographic variation on opposite sides of the Andes while keeping collection under consistent research-station conditions. The dataset used in this chapter was intentionally small and tightly curated: only images that could be labelled confidently in the field and that showed informative healthy tissue or visible early- to mid-stage symptoms were retained for model development. This curation step was the main cleaning stage for the dataset and prioritised label reliability over dataset size. No manual cropping, background removal, or lesion segmentation was applied when preparing the images. Instead, the photographs were kept as field scenes so that the models would be exposed to realistic canopy clutter, illumination variation, and non-diagnostic background structure. Before training, the files were organised into parallel four-class IR and RGB directory structures. Images were resized on the fly to the square input size selected by the optimisation sweep rather than being permanently resampled during dataset construction. In line with the findings of Section 3.4, no additional image normalisation transform was applied. The only routine data transformations used during training were light augmentations intended to improve robustness without distorting disease cues: horizontal flip, Gaussian blur, and random rotation of 0–5 degrees.

5.2.2 Model Development and Optimisation

In PhytNet, several architectural parameters, described below, are determined by a configuration file. This allows the same core model to be resized deliberately for a given dataset rather than treated as a fixed off-the-shelf network. The architecture was therefore not designed in isolation, but as a selective synthesis of ideas drawn from the strongest CNN families reviewed in Chapter 2. Similar to EfficientNet, the network concludes with an adaptive average pooling layer before a fully connected layer, which decouples the classifier from a fixed input size and makes image resolution part of the optimisation problem rather than a hard architectural constraint.

The backbone itself is closest in spirit to a compact ResNet. The PhytNet architecture, illustrated in fig. 5.1, uses residual convolutional blocks so that gradients and low-level image information can pass through the network without the optimisation instability that often accompanies deeper plain CNNs. Figure 5.1a shows the convolutional block of PhytNet. An optimised number of these blocks are applied sequentially in Layer One and Layer Two of PhytNet (fig. 5.1b). The number of channels and kernel size of all convolutional layers are also optimised for a given dataset, but these optimised values are constant within Layers One and Two respectively. The stride values of Layer One and Layer Two are 1 and 2 respectively. This keeps the earliest stage at high spatial resolution so that small lesions, pod edges, and local textural variation are preserved, while the later stage expands the receptive field at lower computational cost. Figure 5.1a also shows two options for the skip connection. The standard identity skip connection on the left is used if the input dimensions of the block are the same as the output dimensions. However, in order to allow for flexibility in kernel size and convolutional dimensions in the optimisation sweep, the convolutional skip connection may adjust the dimensions of the input to match the output without applying an activation function. The convolutional block also automatically pads the skip connection output if necessary to allow for this process. This retains the central advantage of ResNet-style residual learning while providing much more freedom to tune width, depth, and receptive field to the dataset at hand.

PhytNet uses group normalisation in place of batch normalisation, consistent with the considerations reviewed in Section 2.1.14. PhytNet therefore differs from the better-known

state-of-the-art architectures in three main ways. First, unlike ResNet18, EfficientNet, or ConvNeXt, it is intentionally shallow and narrow, with only two main convolutional stages whose depth, width, and kernel size are optimised jointly for the target dataset. Second, except for one max pooling layer, pooling is avoided to preserve fine-grained spatial information rather than compressing it aggressively at an early stage. Third, the choice of group normalisation departs from the default normalisation strategies used by those models. This deviation from the standard ResNet and EfficientNet design was motivated by the small batch sizes and plant-pathology-specific behaviour discussed there [164, 223], while also reflecting the broader ConvNeXt-era lesson that normalisation choice is a first-order architectural decision rather than a minor implementation detail. These choices were made specifically for plant disease detection. In field images, the diagnostically useful evidence is often a relatively small diseased region on a pod surrounded by leaves, branches, soil, specular highlights, and non-diagnostic background structure. Early and mid-stage symptoms may occupy only a small fraction of the frame and may be expressed as subtle changes in colour, texture, or lesion boundary rather than as large, high-contrast objects. A custom architecture for this problem therefore needs enough receptive field to capture pod-scale context, but not so much early downsampling or parameter capacity that it learns canopy background, lighting artefacts, or site-specific biases instead of disease symptoms. PhytNet was designed to sit in that middle ground: small enough to resist overfitting and run efficiently, but flexible enough to tune receptive field and feature width to cocoa-specific symptom scales.

Starting with a basic residual CNN and the intentionally small dataset of 240 IR images, a series of training runs were executed while various configurations were manually tuned and tracked using the Weights and Biases (wandb) platform (San Francisco, California, United States of America (USA)) [217]. These configurations included different activation functions such as rectified linear unit (ReLU) and Gaussian error linear unit (GELU) [108], attention mechanisms including multi-head attention [107] and squeeze-excitation layers [224], dimension reduction such as max pooling, average pooling and adaptive average pooling, convolution block/bottleneck block configurations, stochastic depth [225], dropout [226], batch normalisation [109], layer normalisation [110], group normalisation [111], model depths and dense layer configurations. Any adjustments to the architecture were retained only if they improved the validation F1 score while reducing, or at least maintaining, signs of overfitting. The final model should therefore be understood not as an arbitrary new architecture, but

as a cocoa-specific synthesis that keeps the parts of modern CNN design that improved behaviour on this dataset and removes the parts that mainly added capacity. Table 5.1 shows the results of a post hoc ablation analysis in which we replaced each of the model features shown, such as model layers or activation function, with alternatives.

Table 5.1: Results of an ablation analysis comparing substituted model features relative to the baseline PhytNet model (Figure 5.1). squeeze-excitation (SE) layer = squeeze–excitation.

Model	**Train F1**	**Val F1**
Baseline	0.724	0.679
ReLU → GELU	0.390	0.378
MaxPool → AvgPool	0.348	0.313
Baseline + SE layer	0.357	0.369
GroupNorm → BatchNorm	0.683	0.455
GroupNorm → LayerNorm	0.771	0.640

In addition to performance metrics such as accuracy, F1 and loss, the number of trainable parameters and giga floating-point operations per second (GFLOPS) were also recorded for each model. Choosing a model with a high validation F1 score, comparatively low training F1 and the minimum number of trainable parameters and GFLOPS would help to avoid overfitting and reduce computational cost. This strategy of model choice was also applied in the subsequent optimisation sweep.

Once the architecture was chosen, an optimisation sweep was run using wandb's Bayesian optimisation method to optimise for the validation set F1. The general rationale for Bayesian optimisation and early stopping is reviewed in Section 2.1.3; here it was used to search for a model that performed best on the validation set rather than the training set, thereby discouraging overfitting during selection.

This sweep optimised the kernel size of the middle convolutional layer of each bottleneck block (1:19), number of convolution layer channels (16:128), number of bottleneck blocks in each convolution block (1:4), square image input size (200:500 pixels), learning rate (1^{-6}:1^{-3}), number of output channels (4:10), and beta1 (0.88:0.99) and beta2 (0.93:0.999) values of the AdamW optimiser, which control the exponential moving average of weight updating. These values were selected to facilitate the search for a model that would avoid overfitting, train in a controlled manner while allowing enough stochasticity to properly search the loss landscape, and have an appropriate kernel size to focus its attention on the features of the

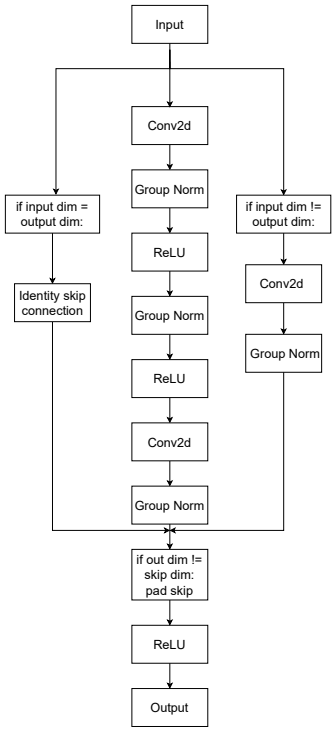
given dataset. The importance of kernel size optimisation is exemplified in ConvNeXt [94].

During the sweep, models larger than 6 GFLOPS or two million trainable parameters were terminated before training. 645 models were trained using one NVIDIA GeForce GTX 1080 Ti graphics processing unit (GPU) and one NVIDIA Quadro P6000 GPU. Each model was trained on a single GPU, taking approximately 4 minutes to train. Early stopping was applied to halt training when the validation loss failed to decrease for 20 consecutive epochs, at which point the checkpoint with the best validation F1 was saved. Simple supervised training was used with AdamW optimisation (weight decay=1^{-4} and eps=1^{-6}). L1 regularisation was added to the cross entropy loss function by summing the value of all model parameters and multiplying that value by a weight of 1^{-5}. The following image augmentations were randomly applied during training: horizontal flip, Gaussian blur and random rotation of 0-5 degrees. We also optimised for the number of output nodes and observed that the best-performing models had 7 or 8 output nodes, despite having only 4 image classes. Potential explanations for this observation are given in the discussion section.

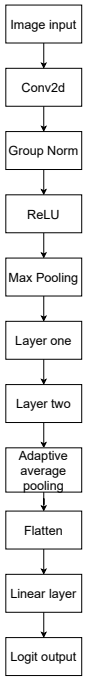
5.2.3 Model Evaluation

To assess the performance of PhytNet, we compared it to four competing architectures: ResNet18, EfficientNet-b0, EfficientNet-V2S and ConvNeXt tiny. Each of these four architectures were trained using the procedure described above with the addition of an optimisation sweep for image input size, learning rate, and beta1 and beta2 values of AdamW. This allowed for a fair comparison between models, giving the competing architectures a chance to avoid overfitting and produce favourable results. To ensure reproducibility, torch backends were set to deterministic, torch random seeds were set to 42 and the data loader random seeds were set based on worker ID. Initial weights were generated using PyTorch default methods rather than using pre-trained ImageNet weights to ensure a fair comparison between models as no ImageNet weights are available for PhytNet yet. Plant dataset-specific pre-trained weights for PhytNet will be produced in a subsequent study.

The performance metrics of the four models were estimated using 10-fold cross-validation, following the principles outlined in Sections 2.2.4 and 2.2.7. While 10% of the data was reserved for validation during model development, the typical 10% test split was not used



(a) PhytNet Convolutional block design



(b) Full PhytNet design

Figure 5.1: Schematic diagram of PhytNet convolutional neural network. Layers one and two are composed of a predefined number of sequential convolutional blocks. The number of channels and kernel size of all convolutional layers should be optimised for a given dataset but these optimised values are constant within layer one and layer two, respectively. The stride values of layer one and layer two are 1 and 2, respectively.

for final evaluation for three reasons: 1) the images are highly consistent in their features, meaning that any small hold-out subset would closely resemble the overall distribution and risk pseudo-replication, 2) the dataset is too small to justify discarding further training data for a single final test split, and 3) cross-validation provides a more conservative and distribution-based estimate of model performance [170].

To compare the validation-score distributions between models more formally, we also carried out pairwise non-parametric statistical tests on the cross-validation results. Mann–Whitney U tests were used to assess whether one model tended to rank higher than another, Kolmogorov–Smirnov tests were used to assess differences in the overall empirical distributions, and the Vargha–Delaney A measure was used as an effect-size statistic to quantify stochastic superiority between model pairs. These tests were applied to the cross-validation validation results as a supplement to, rather than a replacement for, the practical evidence

Table 5.2: Results of 10-fold cross-validation analysis comparing PhytNet to four competing architectures, all trained on the same IR cocoa disease dataset. *n.b.* GFLOPS were calculated using the optimised input size, shown here as pixels^2. GFLOP values reported by the original model authors may differ because of this.

Model	Train F1	Val F1	Train loss	Val loss	GFLOPS	n pixels^2	n parameters
PhytNet	0.60	0.61	0.93	1.17	1.19	285	336,196
ResNet18	0.62	0.65	1.80	1.94	6.16	408	11,178,564
EfficientNet B0	0.87	0.69	2.31	3.16	1.45	424	3,970,656
EfficientNet V2S	0.89	0.71	2.26	3.32	13.23	485	19,913,468
ConvNeXt Tiny	0.41	0.52	1.82	1.88	3.77	212	27,813,508

provided by loss values, computational cost, and Grad-CAM analysis.

Finally, class activation maps were produced with Grad-CAM to inspect the informative features used by each model, following the broader interpretability rationale outlined in Section 2.2.9 and table 2.9. In this chapter, these maps were used specifically to assess overfitting and to catch naive behaviour that is not apparent in summary statistics alone.

5.3 Results

5.3.1 Model Evaluation

Table 5.2 shows that while EfficientNet-b0 and EfficientNet-V2s give a relatively high mean validation F1: 69% (95% CI: 0.64-0.74) and 71% (95% CI: 0.66-0.75) respectively, both have a mean training F1 18 percentage points higher than their mean validation F1, suggesting that they overfit to this data. This is corroborated by their higher validation loss than training loss and by the class activation maps produced using Grad-CAM (fig. 5.2). Figure 5.2 shows that EfficientNet-b0 focuses its attention poorly, except perhaps for the first WBD image. EfficientNet-v2s seems to focus its attention quite well on the BPR images, though in both cases it seems to focus more on the healthy tissue than the disease lesions. Furthermore, EfficientNet-v2s seems to fail completely to focus its attention on disease symptoms, or even the focal tree, in the other images, despite getting most classifications correct. Taken together, the large train–validation gaps, the inflated validation loss, and the biologically unconvincing activation maps constitute strong evidence of overfitting in both EfficientNet variants. This is a textbook definition of overfitting, *i.e.* these models utilise image features

that are consistent in the training set but are unrelated to the actual disease symptoms, potentially leading to poor generalisation to new or unseen data. Additionally, at 13.23 GFLOPS, EfficientNet-v2s is far more computationally expensive than the other models described here, making it inappropriate for rapid classification when used on edge devices. ConvNeXt tiny performed with poor F1 scores on this dataset (Table 5.2), though this is not surprising given its huge number of parameters (27.8 million) and the small size of this dataset. However, ConvNeXT tiny's loss values were similar to those of ResNet18 and, with an optimised input size of 212 pixels^2, its computational cost was measured at a very low 3.77 GFLOPS. However, owing to the low F1 scores, ConvNeXt is not considered in subsequent analyses here.

Pairwise non-parametric comparisons of the cross-validation validation-score distributions supported this interpretation. Mann–Whitney U testing found no statistically significant difference between PhytNet and ResNet18 (p=0.075) but did detect differences between PhytNet and EfficientNet-b0 (p=0.014), EfficientNet-V2s (p=0.002), and ConvNeXt tiny (p=0.002). The Vargha–Delaney A measure was 0.26 for PhytNet versus ResNet18, 0.17 versus EfficientNet-b0, 0.095 versus EfficientNet-V2s, and 0.915 versus ConvNeXt tiny, indicating that the two EfficientNet variants tended to produce higher cross-validation scores than PhytNet, whereas PhytNet tended to outperform ConvNeXt tiny.

The Kolmogorov–Smirnov test similarly suggested that the overall distribution shape of PhytNet was not clearly different from ResNet18 (p=0.418) or EfficientNet-b0 (p=0.052), but was different from EfficientNet-V2s (p=0.012) and ConvNeXt tiny (p=0.012). Together, these results indicate that the validation-score distributions of PhytNet and ResNet18 overlapped substantially, while the EfficientNet variants occupied a somewhat higher-scoring but behaviourally less trustworthy regime. In the context of the strong overfitting evidence from train–validation gaps and Grad-CAM, these higher EfficientNet scores should therefore be interpreted cautiously rather than taken at face value.

PhytNet and ResNet18 had almost perfectly consistent mean training and validation F1 scores, with PhytNet providing the lowest mean training and validation loss values. However, the difference between the training and validation loss values of PhytNet was the third highest in this comparison, which is an indication of slight overfitting. Additionally, as we will now

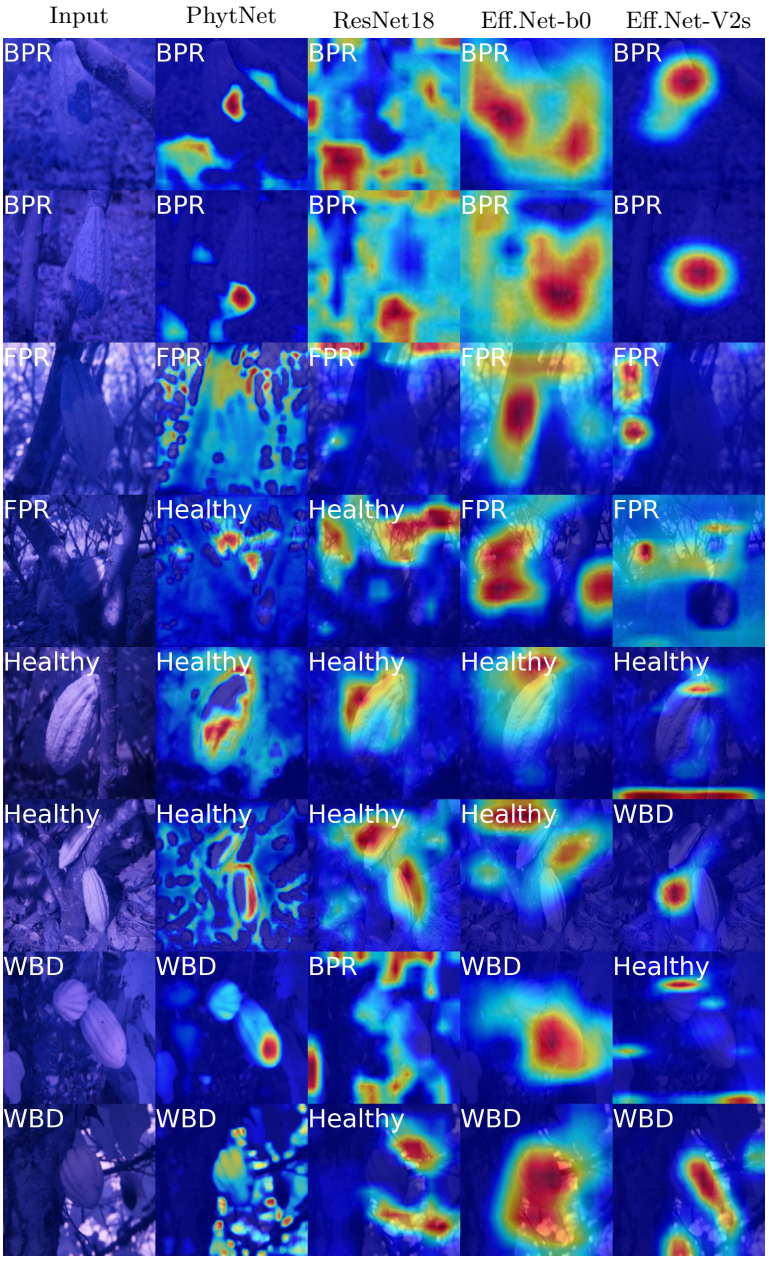
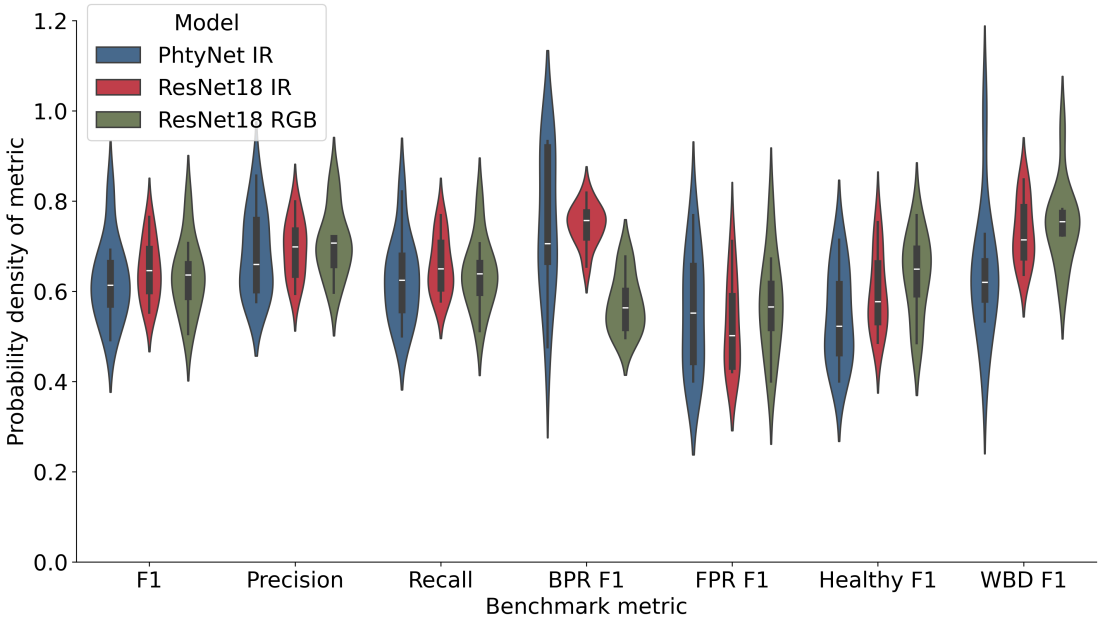
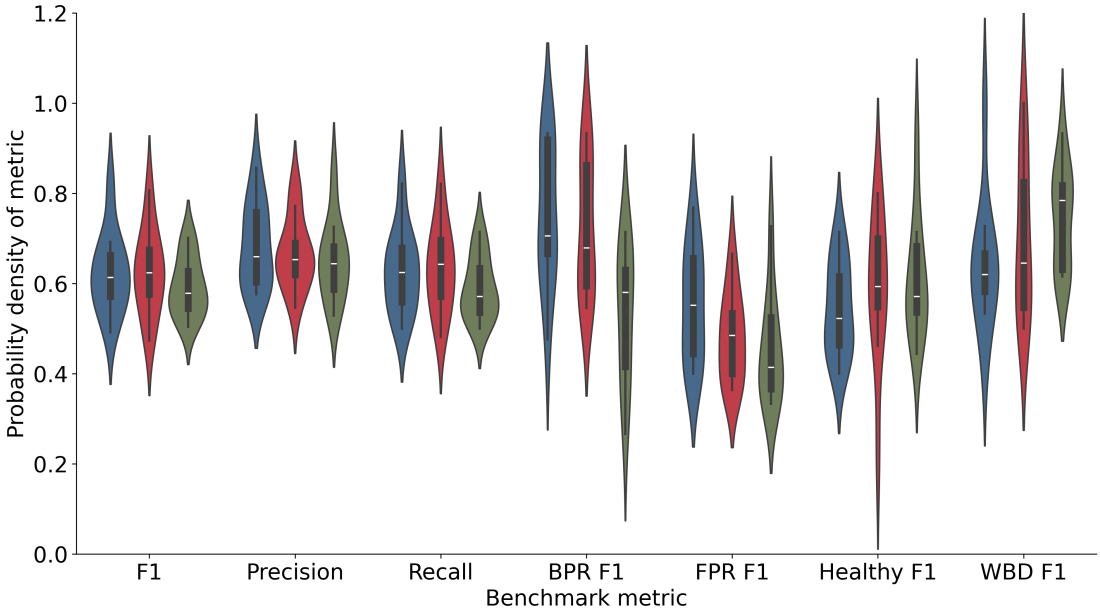


Figure 5.2: Infrared images with class activation heatmaps produced using Grad-CAM and four CNNs. Models used are PhytNet, ResNet18, EfficientNet-b0 and EfficientNetV2 (left to right). The leftmost column shows raw input images with ground truth labels in white, other white labels are predicted by each model.



(a) Training dataset results



(b) Validation dataset results

Figure 5.3: Violin and box plots of 10-fold cross-validation results for PhytNet and ResNet18 trained(a) and validated(b) on infrared or RGB images of cocoa disease. Shown here is the Gaussian density function, medium and interquartile ranges for mean F1, per class F1, precision and recall. PhytNet was trained only on IR data, while ResNet18 was trained on IR or RGB data. The datasets had four classes: Black pod rot, Frosty pod rot, Healthy and Witches' broom disease. n=70 images per class of early to mid-stage diseased or healthy cocoa with a 90%:10% train, validation split.

discuss, these simple metrics are hiding naive behaviour on the part of ResNet18. Figure 5.2 shows that ResNet18 is focusing its attention slightly better than EfficientNet, while PhytNet is focusing its attention on disease symptoms exceptionally well. We also see in both FPR images of Figure 5.2, that all four models seem to show some signs of being "distracted" or "overwhelmed" by the bright sunlight in the images, despite all four models making the correct classification in the first FPR image. However, despite this apparent distraction, PhytNet still shows that it focuses its attention on the cocoa pod or disease lesion.

Figure 5.3 shows, with greater detail than Table 5.2, the results from the cross-validation analysis of PhytNet trained on IR images and ResNet18 trained on IR or RGB images. These two model architectures are compared here as they fit best to the data according to the summary statistics in Table 5.2 and the activation maps in Figure 5.2. ResNet18 was chosen to compare training on the IR and RGB datasets as it is the better-known and tested architecture.

Figure 5.3 shows that the median values per metric are very similar between models, with the exception of PhytNet's slightly higher validation F1 for FPR. ResNet18 seems to have a higher median validation F1 for Healthy but with very long tails for this class and WBD, meaning that its performance on the validation set was highly variable for these classes. Additionally, we see that the distribution of ResNet18 F1 values across classes for the training set is much narrower than in the validation set, with an almost perfect Gaussian distribution for training BPR F1. This indicates some degree of overfitting. *i.e.* ResNet18 is learning the training data very well but struggles to generalise consistently across the validation set.

Comparing the PhytNet training and validation results, the relatively Gaussian shapes of the distributions for F1, precision, and recall suggest that PhytNet is providing consistent and reliable results across the dataset. This consistency is further reinforced by the similarity between training and validation distributions. The wide tails in the PhytNet per class F1 scores are a concern as they indicate that its performance for specific classes is quite variable. However, considering the very small size of the dataset used here (per class train n = 63, val n=7), this variability in model performance is likely due to inconsistencies and noise in the data for specific classes, affecting the model's performance. The identical distributions between the training and validation set for PhytNet suggest that the model is not memorising

the training data but is instead learning genuine patterns that apply to unseen data.

Additionally, the ResNet18 mean training F1 for FPR was 13% **lower** than the mean FPR validation F1. This curious behaviour, in conjunction with the failure of ResNet18 to consistently isolate features of interest in the Grad-CAM analysis and the inconsistent distributions of values between train and validation sets, suggests that ResNet18 is overfitting, if to a lesser extent than the EfficientNet variants.

Figure 5.3 also shows a slight improvement in FPR classification using the IR images over the RGB images. This suggests that some image-level non-visible cues may still be useful for cocoa, particularly for FPR, even though architecture choice remains the main focus of this chapter. In light of the preceding spectroscopy chapter, however, this result should be interpreted cautiously: it is not confirmation that non-visible sensing is robustly superior, but a narrower indication that infrared imagery may retain some practically useful signal even when broader field-spectroscopy evidence is batch-sensitive and inconclusive. We also see in Figure 5.3 that the BPR F1 distribution was markedly lower when using the RGB dataset, while the median WBD F1 was much greater.

5.4 Discussion

5.4.1 Evaluation of CNN Architectures

The EfficientNet variants gave high F1 scores but also showed clear signs of overfitting. Indeed, they provided the strongest overfitting evidence of any models tested in this chapter: large train–validation gaps, higher validation than training loss, and activation maps that frequently failed to isolate disease symptoms or even the focal tree. ConvNeXt tiny, at the other extreme, brought a modern state-of-the-art design but too much capacity for a dataset of this size. In contrast, PhytNet and ResNet18 demonstrated more consistent performance in F1 scores and Grad-CAM analysis. PhytNet performed by far the best in the Grad-CAM analysis, while also having the lowest loss values and least GFLOPS. Taken together, these results suggest that the most suitable starting point for this task was not the largest or newest available architecture, but a compact residual CNN that could be adapted carefully

to the data.

PhytNet was developed from precisely that observation. Architecturally, it is best understood as a plant pathology-specific simplification of the state of the art. From ResNet18 it keeps residual shortcuts, but it departs from standard ResNet by making stage depth, channel width, and kernel size explicitly tunable and by replacing batch normalisation with group normalisation. From EfficientNet it borrows the idea that input resolution and model scale should be matched deliberately to the dataset and it retains adaptive average pooling, but it does not inherit EfficientNet's tendency toward higher-capacity scaling. From ConvNeXt it takes the broader lesson that kernel size and macro-design matter, but it avoids the larger parameter budgets and design assumptions that are better suited to ImageNet-scale problems. The main difference, therefore, is not novelty for its own sake but selectivity: PhytNet keeps the aspects of modern CNN design that improved behaviour on cocoa images and omits those that mainly increased capacity or complexity.

This interpretation is consistent with a broader pattern in plant-disease detection, where residual CNNs often remain strong practical baselines even when newer architectures are available. In cocoa, work by [99] compared ResNet18 and ResNet50 for bean classification and found that the smaller residual model was the stronger option, reinforcing the point that more parameters do not automatically produce better agricultural models. More generally, benchmark-style successes on curated disease datasets, including challenge datasets with relatively clean and abundant imagery [215], do not remove the need to match architecture size and inductive bias to the biological task and to the realities of field data collection. The present results follow that logic closely: PhytNet did not win because it was the most complex architecture tested, but because it was the model whose design assumptions were most proportionate to the scale, noise, and visual structure of cocoa disease imagery.

The effects of bright sunlight may partly explain why the median FPR F1 was relatively low in all models here. As the causative agent of FPR, *M. perniciosa*, is wind dispersed [68] FPR tends to be found above two meters in the canopy. This means that bright light will be more common in these images than in images of the soil-borne BPR pathogen, *Phytophthora spp.* [72]. This may suggest a need for controlled imaging conditions. However, the fact that all four models correctly classified the first FPR image, despite this apparent distraction,

suggests that the models might still leverage other features for classification when bright light is present. While PhytNet focuses on the diseased pods in addition to bright light, the other three models focus on irrelevant features such as the tree trunk, background or only bright light. However, despite this, PhytNet classifies the second FPR image as healthy, possibly because only healthy pod tissue is brightly illuminated.

These differences matter because cocoa disease detection is a narrow biological task carried out under field constraints rather than a generic benchmark problem. The images are few, the backgrounds are cluttered, lighting is variable, and the lesions of interest are often small relative to the full frame. For that reason, PhytNet preserves spatial detail early in the network, constrains parameter count to 336,196 and computational cost to 1.19 GFLOPS, and uses a normalisation strategy that is stable under small-batch training. The Grad-CAM maps suggest that these decisions were not merely computational conveniences: they changed what the model learned, directing attention towards pods and lesions rather than towards background structure.

This does not mean that newer architectures lack promise for plant pathology. Vision transformers, hierarchical transformers such as Swin, updated convolutional families such as ConvNeXt V2, and compact transformer variants such as TinyViT may become highly competitive when stronger pre-training, larger multi-site datasets, or multimodal inputs are available [93, 227, 228, 229]. Their appeal lies in improved global-context modelling and, in some cases, better scaling behaviour. However, those strengths are most likely to pay off when the available data are sufficiently large and diverse to support them. For this cocoa dataset, the results suggest that the immediate opportunity was not to adopt the newest architecture family by default, but to distil the most useful lessons from recent state-of-the-art models into a compact network that could learn reliably from limited, noisy, field-acquired images.

While PhytNet had the lowest mean validation F1 score, it is the only model here that showed almost no signs of overfitting, it performed with great consistency between train and validation sets, and it performed best for FPR images. This latter point is of high potential economic value to cocoa farmers. As such, PhytNet would most likely perform best and most consistently in the field because it was custom designed for the biological and practical

constraints of cocoa disease diagnosis rather than for generic large-scale image classification.

The statistical comparison helps sharpen that conclusion. On validation F1 alone, the EfficientNet variants tended to score higher than PhytNet across folds, and those pairwise differences were significant by Mann–Whitney U. However, PhytNet was not statistically separable from ResNet18, despite requiring far fewer parameters and GFLOPS. This matters because the practical question in this chapter is not simply which model attains the highest average score on a small cross-validation benchmark, but which model gives the most defensible balance of score, computational cost, and biologically credible behaviour. By that criterion, the statistical tests reinforce a cautious reading rather than overturning it: they support treating PhytNet and ResNet18 as broadly comparable on cross-validation performance, while the stronger headline scores of the EfficientNet models should be discounted because they are accompanied by the strongest evidence of overfitting and the least trustworthy feature use.

5.4.2 Optimisation of Output Node Number

During the optimisation sweep, PhytNet performed consistently better with seven or eight output nodes, despite having only four classes in the dataset. In challenging classification problems, having more output nodes than classes offers several potential advantages. Extra nodes may capture nuanced feature representations [230], act as "soft clusters" for variations within classes, or serve as a form of regularisation to improve generalisation through increased complexity of model parameters [231]. The extra nodes could also provide a "catch-all" for unknown or less frequent features, thereby preventing forced guesses and acting to prevent representation collapse, where disparate inputs are mapped to the same point or a very narrow region in the feature space [232]. Additionally, a larger parameter space may smooth the optimisation landscape, a characteristic that is said to make it easier for algorithms to find a good solution [233, 234]. These potential explanations for this curious model behaviour should be explored in future work.

5.5 Summary

PhytNet, while promising, has potential limitations that have yet to be tested. In future studies, we will test if its performance is dataset-dependent and how well it captures intricate patterns in more complex data. Although efficient, its reduced complexity might be a compromising factor, and it could be prone to underfitting in certain scenarios. We will test PhytNet's ability in transfer learning in a future study, though with a larger plant pathology dataset rather than ImageNet, which is irrelevant in this context. PhytNet emerged as a promising candidate model architecture in our study, particularly in its ability to focus its attention exceptionally well on relevant features like cocoa pods and disease lesions. Additionally, PhytNet is approximately 5 times faster than ResNet18 at inference time. The superior attention of PhytNet and the apparent complete lack of overfitting offers unique advantages that could be leveraged for specific applications, such as the localisation of disease symptoms on a tree. For example, in automated fungicide application systems, the ability to accurately pinpoint the location of the disease symptom or pathogen could lead to more efficient and targeted application of fungicides, thereby reducing waste and pollution.

Chapter 6

Advanced Training of Neural Networks for Plant Pathology

> "Animated by truth, but lacking free will, a golem always does exactly what it is told. This is lucky, because the golem is incredibly powerful, able to withstand and accomplish more than its creators could. However, its obedience also brings danger, as careless instructions or unexpected events can turn a golem against its makers. Its abundance of power is matched by its lack of wisdom."
>
> *Richard McElreath, Statistical Rethinking*

6.1 Introduction

This chapter builds on the general discussion of model families, semi-supervised learning, hard-example reweighting, and evaluation given in Chapter 2, on the exploratory filtering

studies in Chapter 3, and on the architecture study in Chapter 5. The relevant prior-work discussion is therefore not repeated here. Instead, the chapter concentrates on whether training procedures can improve generalisation, interpretability, and deployment suitability once the architectural choice has already been narrowed (Sections 2.1.4, 2.1.9, 2.2.7, 2.2.9 and 3.3).

This chapter addresses the practical question posed by **RQ3/H3** in Section 1.10: once an architecture is chosen, how much can training strategy improve real-world performance on a difficult plant-pathology problem? In doing so, it provides a direct test of whether advanced training procedures, including semi-supervised learning, a non-cocoa class, and dynamic focal loss (DFLoss), improve fit to difficult real-world cases and reduce overfitting. Because those interventions are evaluated not only by summary performance but also by attention behaviour, runtime, and transfer to an independent acquisition pipeline, the chapter also contributes evidence toward the overarching **RQ0/H0** and the deployment-oriented interpretability question in **RQ4/H4**.

To answer this, we use a cocoa image dataset covering healthy trees and trees infected with black pod rot (BPR) (*Phytophthora palmivora (Butler)* [62]), witches' broom disease (WBD) (*Moniliophthora perniciosa (Stahel)* [235]) and frosty pod rot (FPR) (*Moniliophthora roreri (Ciferri)* [236]) at different symptom stages. This is a difficult classification problem because these diseases are visually heterogeneous, symptoms may be subtle, partially occluded, or confounded by lighting and background clutter, and the model must separate genuine pathology from incidental field cues. Cocoa diseases have been studied for decades because they threaten a crop of global economic and social importance, and because black pod, witches' broom, and frosty pod rot each have distinct epidemiology, symptom expression, and management challenges. [237, 21, 238] Historically, diagnosis has depended on field inspection and expert judgement, but symptom variability and hidden infection stages make this unreliable at scale, especially when background clutter and sensor variation are present. [218] Modern computer vision (CV) changed this landscape after convolutional neural networks became dominant in image recognition, with residual and efficient architectures such as ResNet showing that strong performance can be achieved without excessive architectural complexity. [6].

For agricultural imaging, however, model performance must be judged alongside robustness and interpretability, because real deployments face more distribution shift and label ambiguity than benchmark datasets. [89, 205] This is why semi-supervised learning is attractive here: it can exploit difficult but informative examples that a human can label only with effort, while limiting the need to scale the architecture simply to absorb noise. [152]

The specific contribution of this chapter is therefore to test whether carefully constrained training procedures can improve model behaviour beyond what was achieved through architecture choice alone in Chapter 5. To that end, we compare semi-supervised training, the addition of a non-cocoa class, and a novel DFLoss, and we evaluate these variants using the multi-criteria framework introduced in Section 2.2.7 and table 2.9.

A further emphasis is placed on generalisation. Alongside the training and validation splits, the chapter uses a genuinely independent test set contributed by Fairfield Vision Ltd., York, UK, gathered in the same cocoa-growing regions of Ecuador. This allows the chapter to separate strong validation performance from performance that transfers to a genuinely new acquisition pipeline, and therefore to distinguish optimisation to the development data from more credible evidence of real-world robustness.

The results and discussion are organised around three practical questions: first, whether semi-supervised learning helps the models make better use of difficult images without overfitting; second, whether adding a non-cocoa class improves robustness to real-world confounders; and third, whether DFLoss improves generalisation consistently or only for some model capacities and architectures. The answers are interpreted through summary metrics, per-class behaviour, relabelling rates, gradient-weighted class activation mapping (Grad-CAM) analysis, runtime measurements, and independent-test-set performance, so that model quality is assessed not only by accuracy but also by practicality for deployment.

6.2 Methods

6.2.1 Data Collection

Four datasets were used in this study; 1: the forestry and arable images from Google and Bing (FAIGB) dataset, a rigorously curated dataset of 31,225 healthy and diseased images of forestry and arable plants assembled from Google and Bing for this work and refined through automated filtering and manual review [223], described in Section 2.2.2. 2: a dataset of 7,220 red, green, and blue (RGB) images of diseased and healthy cocoa trees collected across Ecuador and augmented with images from the web. We collected the novel second dataset for this study and will now describe it in detail. 3: a dataset of 1,963 non-cocoa plant images randomly sampled from the non-cocoa portion of the FAIGB dataset, described in Section 2.2.2, used to create a non-cocoa class for training and testing. And 4: a test dataset of 4433 RGB images of classes BPR: 1145, FPR: 525, Healthy: 1199, NonCocoa: 953 and WBD: 611. This test set was contributed to this project by Fairfield Vision Ltd., York, UK. It was gathered using the Phoenix polarisation camera with Sony's IMX264MZR/MYR polarised CMOS sensors across 2025 and 2026 in the same regions of Ecuador as the training and validation sets. Thus, it represents a consistent but truly independent test set on which to test these models' generalisation. The first two datasets were used for training and validation, while the third dataset was used to create a non-cocoa class for training and validation. The fourth dataset was used as an independent test set to evaluate model generalisation.

The locations of the sites where these images were gathered are shown in fig. 6.1, and the Global Positioning System (GPS) coordinates for each site are shown in Table 6.1. These sites were not pre-selected through a formal sampling design; rather, they represent all farms and research stations made available for access by Barry Callebaut and Instituto Nacional de Investigaciones Agropecuarias (INIAP) during field data collection. This set of available sites spanned all three cocoa-growing regions of Ecuador, which at the time was the largest global producer affected by all three focal diseases considered in this chapter. The three regions covered were: (1) the dry coastal region (represented here by Ceracita and Calceta), (2) the central region between the Cordillera Chongón-Colonche and the Andes (wet, warm, and humid), and (3) the region between the Andes and the Amazon (very wet, warm, and

Table 6.1: Latitude and longitude of data collection locations

Name	Latitude	Longitude
Riviera cacao	-2.083064	-79.286160
Hacienda San Jose	-1.888611	-79.489074
Naranjal	-2.697633	-79.643309
Near Riviera	-2.064715	-79.244245
INIAP – Litoral Sur	-2.256634	-79.644175
Riviera cacao – associate	-2.677806	-79.650140
Riviera cacao Tambo	-1.980887	-79.258232
INIAP – Pichiligue	-1.075552	-79.492659
Patricia Pilar	-0.539016	-79.366570
Calceta	-0.850179	-80.180915
INIAP – Amazon	-0.340658	-76.874867
Ceracita	-2.330835	-80.270115

humid), represented here by INIAP-Amazon. We gathered images at research stations and farms of varying sizes across these regions. The smallest farms had 5–10 trees while the largest farms had 200,000 to 500,000 trees.

Data collection was guided in part by a ResNet18 convolutional neural network (CNN) that we trained on images scraped from the internet and which was deployed using a Google Pixel 3a smartphone (Google, Mountain View, California, United States of America (USA)). This allowed us to identify and gather images of the sort that this initial prototype model misclassified. For example, we found that images of young cocoa leaves that were scraped from the internet were mostly infected with witches' broom disease. As a result, this initial model tended to classify young cocoa leaf images as having this disease so we bolstered the data set by gathering many images of healthy young cocoa trees and leaves. This data set consists of four classes; black pod rot (BPR; *Phytophthora palmivora* (Butler), witches' broom disease (WBD; *Moniliophthora perniciosa* (Stahel), frosty pod rot (FPR; *Moniliophthora roreri (Ciferri)*) and healthy cocoa. In this case, "healthy" cocoa constituted all trees not infected by disease and so included abiotic stressors such as damage from solar radiation or mechanical damage from overzealous pruning. A fifth class in this dataset was the "non-cocoa" class, which was created by sampling randomly from the non-cocoa images of the FAIGB dataset with a similar frequency to the other four classes. Cocoa images from the web-scraped dataset were also added to the Ecuador dataset in the corresponding classes. In total these five classes had the following number of images; BPR: 1752, FPR: 1907, WBD:

Figure 6.1: Farm and research station sites of cocoa image data collection across Ecuador. Note that these sites span the three distinct cocoa growing regions of Ecuador: 1) the dry west, coastal region represented here by the two westernmost points - Ceracita and Calceta, 2) the humid and high-yielding valley between the Cordillera Chongón-Colonche and the Andes mountains. These farms and research stations are shown by the eight centre points, and 3) the region between the Andes and the Amazon rain forest, known for high yields and genetic diversity of cocoa trees. The research station where this data was gathered is shown by the easternmost point - INIAP Amazon.

Table 6.2: Frequency of images in the focal cocoa dataset by class and camera manufacturer/source

Class	Google	Olympus	Samsung	Vivo	Web	Total
BPR	310	202	1092	81	67	1752
FPR	658	325	735	103	86	1907
Healthy	615	364	675	74	213	1941
WBD	326	674	577	0	43	1620
NotCocoa	0	0	0	0	**1963**	1963
Total	1909	1565	3079	258	2372	9183

Black pod rot (BPR, *Phytophthora spp.*); frosty pod rot (FPR, *Moniliophthora roreri*); witches' broom disease (WBD, *Moniliophthora perniciosa*). Images in the Vivo subset originated from the "Enfermedades cacao" dataset [169].

1620, Healthy: 1941 and NotCocoa: 1963. The frequencies of these classes, divided into disease categories diagnosed in the field and the corresponding image sources, are summarised in Table 6.2. All images not scraped from the web or sampled from the FAIGB dataset were taken in the field in Ecuador with either a Google Pixel 3a, Samsung Galaxy Xcover 4 (Suwon-si, South Korea), Samsung Galaxy J3 or an Olympus OMD-EM4 camera (Shinjuku, Tokyo, Japan). While most images from the field were taken by the present authors, many were taken by additional volunteers. This use of a variety of cameras and photographers helped to control for the effect of camera and sensor design, different post-image capture processing techniques applied by the devices and any biases of the photographers.

Smaller cocoa plantations were searched for disease in a grid pattern, while in larger plantations, the search for disease was guided by the farmers. Images were labelled in the field as they were collected. In cases where a diagnosis was ambiguous, pods were dissected to confirm the diagnosis after taking photos and/or the advice of experienced people was sought.

In addition to capturing as many photos of diverse symptoms as possible, we also worked to include a variety of backgrounds as well as potentially misleading features such as machete damage, insect or small animal damage, and bird excrement on cocoa pods that may confuse a CV model deployed in the field.

6.2.2 Semi-Supervised Training

Building on the conservative semi-supervised framing reviewed in Section 2.1.9, we take the training procedure developed in Chapter 5 and wrap it in a semi-supervised learning loop as shown in algorithm 1.

Algorithm 1 Semi-supervised learning loop

```
train(model)
for ToLabelDataSet in [DifficultImages, UnsureImages] do
    while nRelabledImages > 0 do
        previous F1 score (F1) = eval(model)
        for image in ToLableDataSet do
            classify image
            if prediction = label then
                label image & add to train data
            end if
        end for
        train(model)
        new F1 = eval(model)
        if new F1 < previous F1 then
            end training
        end if
    end while
end for
```

During data collection, as well as labelling images by the four classes (BPR, FPR, Healthy and WBD), images were also labelled as "Easy" or "Difficult" to classify or diagnose or "Unsure" of classification. This allowed for the semi-supervised learning approach to be applied where the images were considered difficult to classify or where human classification using only the image was unsure. Such images were added to the training data by the model only if they contained informative features. The iterative semi-supervised learning process was allowed to continue until either no more images were relabelled or the validation F1 of the model failed to increase after re-training with the latest batch of relabelled images. Originally the labels "Early" and "Late" stage disease development were to be used for this purpose. The intention here was that the model would progressively learn to detect earlier and earlier disease symptoms. However while this method may have worked for BPR, which presents with small brown external lesions that turn into large brown external lesions over time, it would be unrealistic to expect a CV model to understand how late-stage FPR or WBD symptoms relate to early-stage symptoms as both of these diseases have an invisible

intermediary stage that occurs within the pod or branch tissue. This point is illustrated in fig. 6.2. Taking FPR as an example, the almost imperceptible bump shown in fig. 6.2 B is symptomatic of initial infection. This bump subsequently disappears as the pod continues to grow, seemingly in good health, while the inside is being destroyed by the fungus. The fungus then reappears on the outside of the pod as shown in fig. 6.2 E, engulfing it in irregular brown lesions and then with white mycelium. In developmental psychology and video-based object tracking, the phenomenon that may allow a person or model to reason about the location or state of invisible objects is referred to as object permanence and has proved to be exceedingly difficult for neural networks to model [239]. Additionally, early-stage WBD can be just as easy to detect as late-stage WBD. However, using the Easy-Difficult labels allowed for small or obscured but late symptoms to be sensibly labelled as difficult, for early but obvious symptoms to be labelled as easy and for the apparently healthy side of a pod to be labelled as unsure if it had a prominent lesion on the other side. The disease state was labelled in the field, but the difficulty label was applied in the lab. Upon visually reviewing each image, the easy label was applied to images of pods or trees with clear and easily diagnosable symptoms. The difficult label was applied if the disease state could be diagnosed with confidence, but symptoms were small, distant, slightly obscured from view or atypical. The unsure label was applied to images that could not be diagnosed with confidence by a human from the image. Such an image might not allow for diagnosis because; 1) the diagnosis in the field was only confirmed after taking the photo of the intact pod and subsequently dissecting it, 2) the visible symptoms used for diagnosis in the field were intentionally obscured from view to create images of this class, or 3) the symptoms were present and visually clear but cryptic. This method should allow for informative patterns found by the model in the easy images to be reinforced as training continues and prevent the model from overfitting to data that does not contain informative features. By encouraging such deductive reasoning, we aim to allow the CV model to learn features for disease detection that humans do not use, without forcing it to find spurious correlations.

6.2.3 Dynamic Focal Loss

The general rationale for focal loss (FL) and hard-example reweighting is reviewed in Section 2.1.4. Here we introduce DFLoss (DFLoss), a novel extension developed here for biolog-

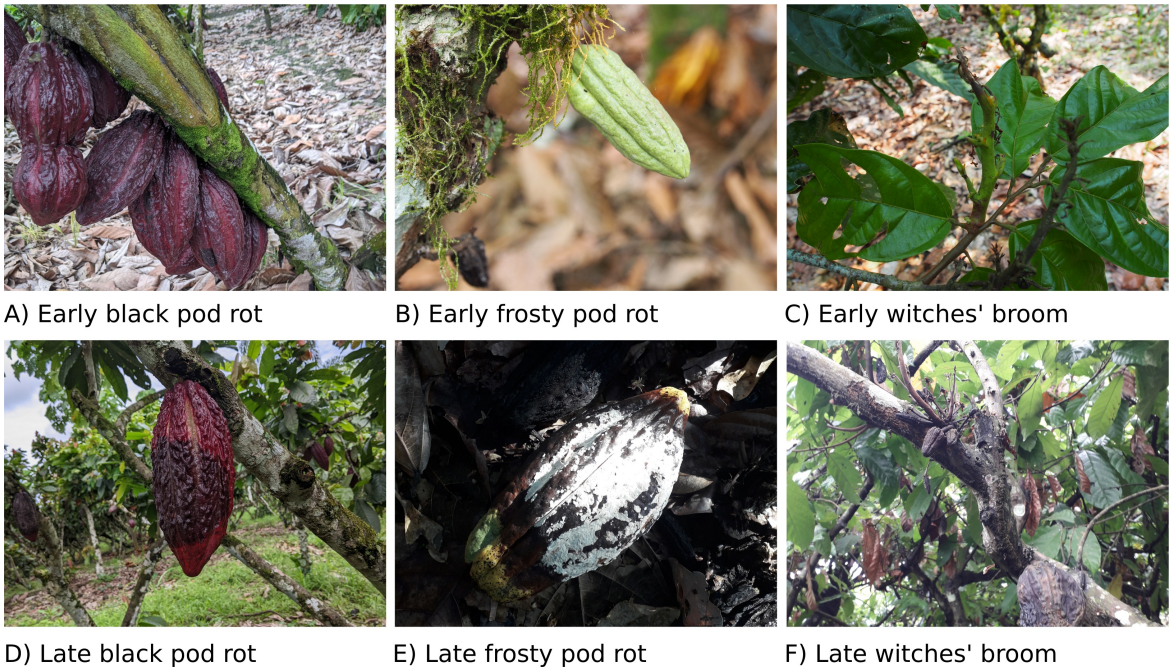


Figure 6.2: Visual progression of three cocoa diseases. Examples show early and late symptoms, demonstrating the temporal discontinuity in disease symptom progression, especially in the case of frosty pod rot.

ically heterogeneous image data in which difficulty varies substantially between observations even when class frequencies are relatively balanced. Using the same notation as Equations (2.1) to (2.3), the key difference between FL and DFLoss is an additional measurement of the difficulty of each observation that is made empirically and then combined with the probability of an incorrect prediction. As such, DFLoss not only encourages a model to focus on observations for which its internal confidence is most awry but, throughout training, increasingly forces the model to focus on observations that it has objectively struggled to classify.

In DFLoss, a dictionary of uniform weights for each observation is generated and stored at the beginning of training. Beginning with the second epoch, when the model fails to classify an image correctly, the stored weight for that image is increased by a previously optimised amount (delta). For a batch of images, this weight is summed and applied to the loss value in place of the gamma smoothing value in FL. This forces the model to focus its efforts on classifying the difficult observations correctly, regardless of class. When the probability of an incorrect prediction approaches zero, the effect of this additional weight is lessened and when an observation is classified correctly, DFLoss reverts to simple cross-entropy loss.

This second point is not true of FL. In DFLoss, an image may gain a higher weight during training because its informative features are less frequent in the training data or because those features are less apparent. By contrast, FL weights an image statically based on class and dynamically based on the pseudo-probability distribution calculated by cross-entropy relative to the label distribution. As such DFLoss should also achieve the same effect as FL for class imbalance without the need for an optimised class weight. We will test the effect of DFLoss with the unbalanced FAIGB dataset in future work. In the present work, we test the efficacy of DFLoss against cross-entropy loss and a custom implementation of FL defined for a balanced multi-class use cases as shown below.

By contrast, in DFLoss γ is a dynamically measured weight based on the difficulty that the model has predicting the class of a given observation. DFLoss is defined as follows.

With cross-entropy loss still defined as cross-entropy loss (CE), the dynamic adjustment based on prediction correctness and weighting by γ is defined as follows:

- For each sample i, a unique weight w_i is updated based on whether the prediction is correct or incorrect. Incorrect predictions result in an increased weight w_i, starting from 1 and incremented by δ for each incorrect prediction of the sample. δ should be optimised as a hyperparameter.
- The aggregated γ is the sum of weights w_i for all samples in the batch that are incorrectly classified.

DFLoss is then calculated as:

$$\text{DFLoss} = \frac{1}{N}\sum_{i=1}^{N}\left(1 + (1-p_t)^{\gamma_i} \cdot \mathbf{1}_{\{pred_i \neq target_i\}}\right) \cdot \text{CE}(y_i, \hat{y}_i) \tag{6.1}$$

where:

$$\gamma_i = \sum_{j=1}^{N} w_j \cdot \mathbf{1}_{\{pred_j \neq target_j\}}, \tag{6.2}$$

$$\mathbf{1}_{\{pred_i \neq target_i\}} = \begin{cases} 1 & \text{if } pred_i \neq target_i \\ 0 & \text{otherwise} \end{cases}, \tag{6.3}$$

N is the total number of samples in the batch and 1 is added to the dynamically calculated weight to ensure that CE is unaffected when all predictions in a batch are correct.

6.2.4 Addition of a Non-Cocoa Class

In addition to the images of cocoa collected from Ecuador and scraped from the web, models here were also trained and tested with and without an additional "non-cocoa" class. These 1,963 images were randomly sampled from the non-cocoa portion of the FAIGB dataset, described in Section 2.2.2, in roughly equal frequency to the other four classes. Any cocoa images present in FAIGB were manually filtered out of this subset before sampling. The objective of including this class was to prevent the model from overfitting to features that are not unique to cocoa or a specific cocoa disease. By exposing the model to such a wide variety of non-cocoa plant images that are similar to those it may encounter in a real-world application, the model is allowed to learn more robust correlations. For example, while bare soil does equate to an increased probability of BPR, because *Phytophthora spp.* is soil-borne and splash-dispersed, it is not inherently symptomatic of BPR. Equally, an image showing the sky does not equate to FPR just because *Moniliophthora roreri* is wind-dispersed and is typically found above 2 meters in the canopy. This additional class should bolster the already robust cocoa dataset and help prevent the model from learning spurious correlations.

6.2.5 Grad-CAM Analysis

The role of activation-map inspection in model evaluation is discussed in Section 2.2.9 and table 2.9. In this chapter, we converted that general rationale into a qualitative Grad-CAM score used to compare models and training variants in addition to loss, accuracy, and F1. This score was derived by manual visual assessment of class activation maps from Grad-CAM (fig. 6.3) for each model and training procedure and assigning a score of 0, 0.5 or 1 to each image. To produce the activation maps, an arbitrary number of images (in this case, 40) were randomly sampled from the difficult cocoa images, which here serves as a holdout test set. If a model correctly classified all images and focused its attention on relevant image features, it would score 1 averaged over the 40 images. However, if the model classifies an image correctly but fails to focus its attention appropriately, it will score 0 for that image,

as this would be a strong indicator of overfitting. If the model misclassifies an image but the classification and attention displayed appear sensible, it would score 0.5 for that image, as, although it is a sub-optimal classifier, it has not overfitted. For example, the image in fig. 6.3 (A) was scored 0, 0.5, 0.5, 0.5. In this case, the fully supervised model scored 0 as it failed to correctly label the image or focus its attention on relevant features. However, the other three variants scored 0.5 as although they failed to detect the very small BPR lesion at the base of the pod, they assigned a Healthy label because they focused their attention on the healthy tissue of the cocoa pod. The image in fig. 6.3 (B) was scored 0.5, 0, 1, 0 because the fully supervised model focused its attention well but failed to classify the image as FRP, which was evident due to the obvious malformation of the pod. Whereas the +non-cocoa variant was able to correctly classify the image and focus its attention appropriately, and so scored 1 for that image. The Semi-supervised and +DFLoss-trained models, however, misclassified the image and failed to focus their attention well, so scored 0 for this image.

6.2.6 Architecture Choice

The architecture choice in this chapter is constrained by the prior-work review in Section 2.1.11 and by the empirical results of Chapter 5. That earlier chapter showed that PhytNet, a lightweight thesis-specific CNN, and ResNet18, a robust residual benchmark, offered the strongest balance of performance, interpretability, and compute efficiency on these specialised datasets [240].

For that reason, the present chapter does not repeat a broad architecture comparison against EfficientNet, ConvNeXt, or larger transformer-style alternatives. Instead, it asks a narrower question: given two architectures already justified by prior work, which training procedures best improve their ability to generalise, attend to relevant features, and remain practical for agricultural deployment?

6.2.7 Additional Comparative Experiments

To evaluate whether the effect of DFLoss depended on model capacity, we ran an additional predefined comparison across three optimised PhytNet variants: PhytNet67k, PhytNet183k,

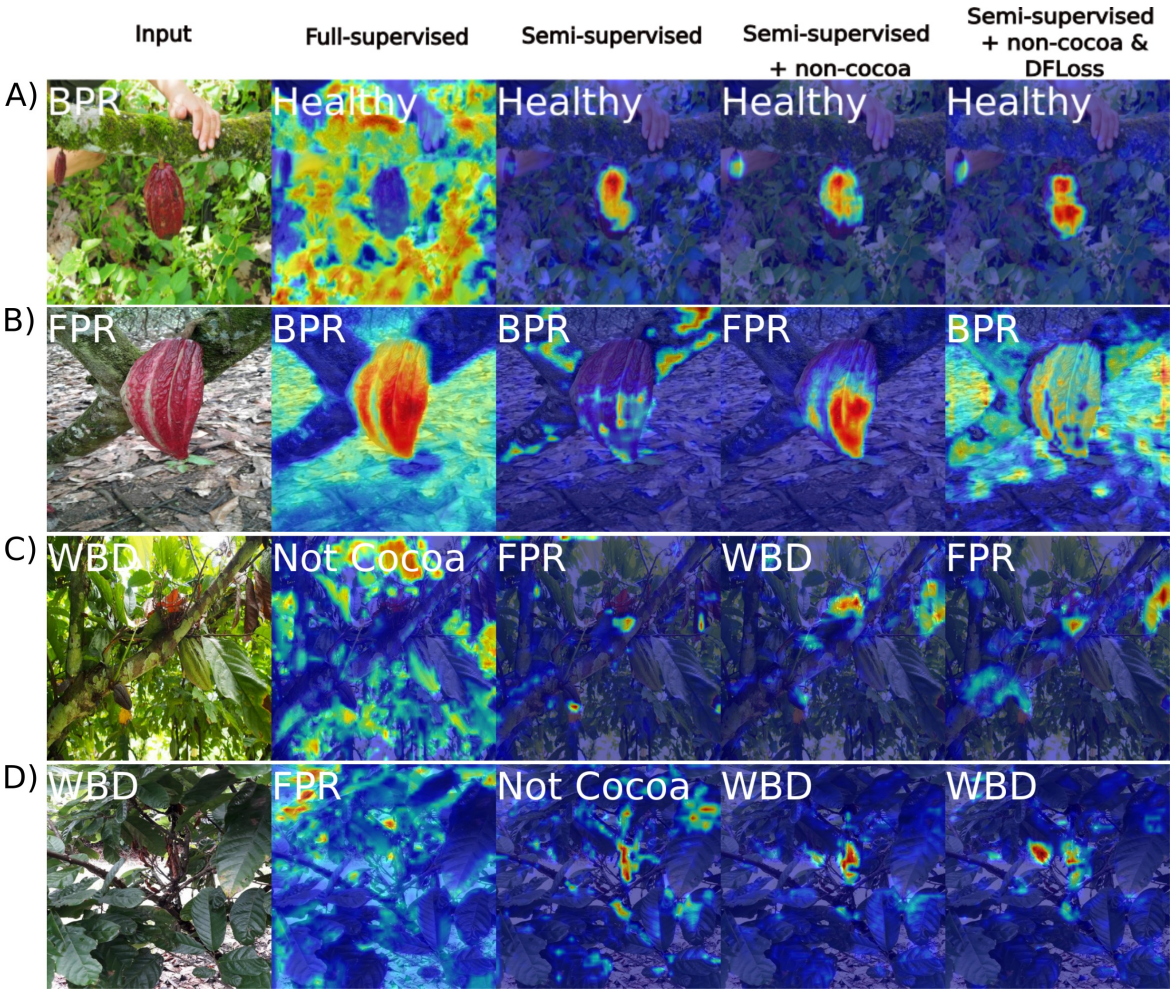


Figure 6.3: Example Grad-CAM class activation maps with scores for PhytNet trained with four of the model training procedures shown in Table 2. The Grad-CAM scores shown range from 0 to 1 and represent both how well each model classified the image and how well it focused its attention on relevant features. (A) Fully supervised: 0, semi-supervised: 0.5, semi-supervised+Non-cocoa: 0.5, semi-supervised+Non-cocoa & DFLoss: 0.5; (B) 0.5, 0, 1, 0; (C) 0, 0, 1, 0; (D) 0, 0, 1, 1. For example, in the top image, all models classified the photo as healthy and focused their attention on healthy pod tissues but failed to detect the black pod rot lesion. BPR, black pod rot; FPR, frosty pod rot; Healthy, healthy cocoa; WBD, witches' broom disease.

and PhytNet332k (table 6.9). These model names refer to the number of trainable parameters. Each variant was assessed under the same semi-supervised + non-cocoa training pipeline, with cross-entropy loss and DFLoss compared directly.

6.2.8 Runtime Evaluation Protocol

Runtime evaluation was conducted for ResNet18 and PhytNet183k using each model's optimised input configuration for this dataset. We reported frames per second (FPS) on graphics processing unit (GPU) and central processing unit (CPU), GPU utilisation, peak video random-access memory (VRAM) use, number of parameters, and giga floating-point operations per second (GFLOPS). Measurements were collected with batch size 100 on an Nvidia H100 80GB GPU and an AMD EPYC 7643 48-core CPU.

6.3 Results

This section reports the observed outcomes of the predefined experiments; interpretation of causes and implications is reserved for Section 6.4. In addition to the train and validation splits, Tables 6.3 to 6.8 report results on a genuinely independent test set for the assessment of model generalisation. This test set provides a more rigorous assessment of model generalisation than the validation set, and the test-set results are a primary focus of interpretation in this chapter, with validation and Grad-CAM results used to support that interpretation.

6.3.1 Semi-Supervised Learning and the Non-Cocoa Class

As shown in Tables 6.3 to 6.5 and illustrated in Figure 6.4 (A), semi-supervised learning improved PhytNet relative to the fully supervised baseline on both the validation and independent test sets, raising mean F1 from 0.482 to 0.692 on validation and from 0.292 to 0.349 on test. Adding the non-cocoa class increased these scores further to 0.755 and 0.377, respectively, and raised the percentage of difficult images relabelled from 27.54% to 77.93%. Among the PhytNet variants in Tables 6.3 to 6.5, the best independent test mean F1 was obtained by semi-supervised + non-cocoa + FL, which reached 0.384, although the semi-supervised + non-cocoa model retained the highest test precision (0.419) and test WBD F1 (0.176).

Figure 6.4 (B) and Tables 6.6 to 6.8 show a clearer divergence between the validation and independent test results for ResNet18. Semi-supervised learning alone gave the strongest test performance, with accuracy 0.562, mean F1 0.549, test BPR F1 0.732, and Healthy test F1 0.584. By contrast, adding the non-cocoa class or DFLoss produced much stronger validation scores but did not improve overall test-set F1, although the non-cocoa class reached validation and test F1 values of 0.990 and 0.695, respectively. This mismatch indicates that validation performance alone would overstate the gains of the more complex ResNet18 training variants on a genuinely independent dataset.

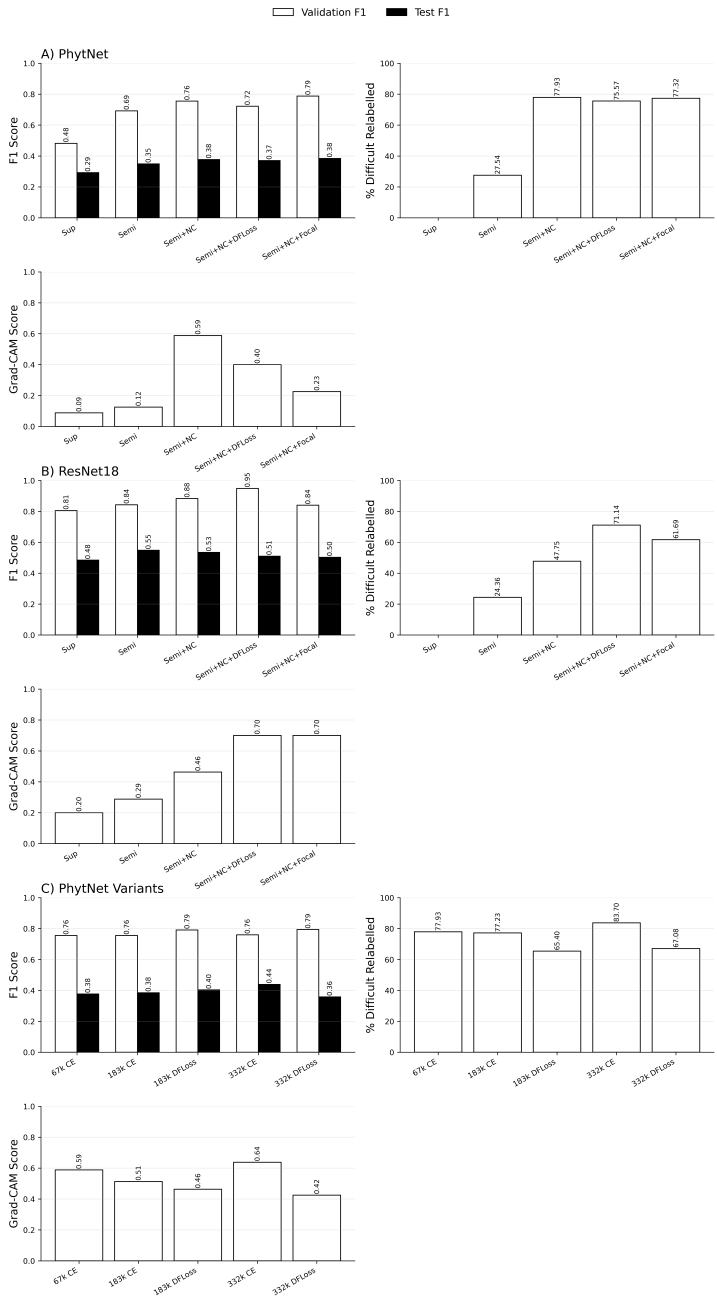


Figure 6.4: Comparison of validation and independent test F1 scores, Grad-CAM scores, and the percentage of difficult images relabelled during semi-supervised training. White bars show validation F1 and black bars show test F1. Relabelling is omitted for fully supervised models. (A) PhytNet improves with semi-supervision and inclusion of the non-cocoa class, though gains reduce on the test set. (B) ResNet18 shows consistently higher performance, with further improvement from DFLoss and smaller validation–test gaps. (C) Increasing PhytNet capacity has limited impact, and DFLoss provides minimal F1 gains while reducing relabelling.

Table 6.3: Overall performance metrics of PhytNet across training variants. Train (T), validation (V), and test (Te) values are shown for loss, accuracy, precision, recall, and F1 score. Bold text indicates values of particular interest.

Variant	Metric	T	V	Te
FullSupervised	Loss	1.222	1.210	1.603
	Accuracy	0.445	0.480	0.291
	Precision	0.470	0.524	0.328
	Recall	0.445	0.480	0.291
	F1 Score	0.449	0.482	0.292
SemiSupervised	Loss	0.597	0.849	1.773
	Accuracy	0.757	0.688	0.345
	Precision	0.787	0.709	0.383
	Recall	0.757	0.688	0.345
	F1 Score	0.763	0.692	0.349
SemiSupervised +NotCocoa	Loss	0.422	0.692	2.604
	Accuracy	0.834	0.756	0.362
	Precision	0.838	0.760	**0.419**
	Recall	0.834	0.756	0.362
	F1 Score	0.834	0.755	0.377
SemiSupervised +NotCocoa, DFLoss	Loss	0.522	0.708	2.363
	Accuracy	0.750	0.720	0.363
	Precision	0.782	0.729	0.410
	Recall	0.750	0.720	0.363
	F1 Score	0.750	0.722	0.371
SemiSupervised +NotCocoa, Focal Loss	Loss	0.407	0.579	**1.905**
	Accuracy	0.840	0.780	**0.388**
	Precision	0.864	0.815	0.411
	Recall	0.840	0.780	**0.388**
	F1 Score	0.841	0.788	**0.384**

Table 6.4: Per-class train (T), validation (V), and test (Te) F1 scores for PhytNet across training variants. Bold text indicates values of particular interest.

Variant	BPR (T/V/Te)	FPR (T/V/Te)	Healthy (T/V/Te)	NotCocoa (T/V/Te)	WBD (T/V/Te)
FullSupervised	0.582/0.654/0.450	0.352/0.375/0.244	0.451/0.463/0.255	N/A	0.380/0.418/0.110
SemiSupervised	0.770/0.725/**0.554**	0.671/0.584/**0.259**	0.808/0.742/0.294	N/A	0.776/0.690/0.148
SemiSupervised +NotCocoa	0.779/0.727/0.501	0.672/0.564/0.186	0.828/0.737/0.353	0.985/0.941/0.492	0.806/0.682/**0.176**
SemiSupervised +NotCocoa, DFLoss	0.774/0.714/0.456	0.556/0.375/0.190	0.815/0.714/0.360	0.931/0.980/**0.554**	0.381/0.632/0.102
SemiSupervised +NotCocoa, Focal Loss	0.788/0.800/0.488	0.706/0.619/0.192	0.873/0.735/**0.394**	0.982/0.939/0.512	0.696/0.800/0.136

Table 6.5: Percentage of difficult and unsure images relabelled during semi-supervised learning, and Grad-CAM score for each PhytNet training variant. Grad-CAM score ranges from 0 to 1 and indicates how well the model focused on informative features. Bold text indicates values of particular interest.

Variant	% Difficult	% Unsure	Grad-CAM score
FullSupervised	N/A	N/A	**0.875**
SemiSupervised	27.54	**30.98**	0.125
SemiSupervised +NotCocoa	**77.93**	40.66	0.588
SemiSupervised +NotCocoa, DFLoss	75.57	43.51	0.400
SemiSupervised +NotCocoa, Focal Loss	77.32	56.95	0.225

Table 6.6: Overall performance metrics of ResNet18 across training variants. Train (T), validation (V), and test (Te) values are shown for loss, accuracy, precision, recall, and F1 score. Bold text indicates values of particular interest.

Variant	Metric	T	V	Te
FullSupervised	Loss	0.015	0.667	2.276
	Accuracy	0.996	0.805	0.509
	Precision	0.996	0.809	0.517
	Recall	0.996	0.805	0.509
	F1 Score	0.996	0.805	0.485
SemiSupervised	Loss	0.003	0.774	2.110
	Accuracy	0.999	0.843	**0.562**
	Precision	0.999	0.844	0.543
	Recall	0.999	0.843	**0.562**
	F1 Score	0.999	0.843	**0.549**
SemiSupervised +NotCocoa	Loss	0.007	0.485	2.024
	Accuracy	0.999	0.883	0.542
	Precision	0.999	0.890	0.549
	Recall	0.999	0.883	0.542
	F1 Score	0.999	0.883	0.535
SemiSupervised +NotCocoa, DFLoss	Loss	0.00019	0.160	2.301
	Accuracy	1.000	0.950	0.530
	Precision	1.000	0.951	**0.560**
	Recall	1.000	0.950	0.530
	F1 Score	1.000	0.949	0.510
SemiSupervised +NotCocoa, Focal Loss	Loss	0.041	0.445	**1.790**
	Accuracy	1.000	0.840	0.543
	Precision	1.000	0.843	0.546
	Recall	1.000	0.840	0.543
	F1 Score	1.000	0.840	0.503

Table 6.7: Per-class train (T), validation (V), and test (Te) F1 scores for ResNet18 across training variants. Bold text indicates values of particular interest.

Variant	BPR (T/V/Te)	FPR (T/V/Te)	Healthy (T/V/Te)	NotCocoa (T/V/Te)	WBD (T/V/Te)
FullSupervised	0.997/0.795/0.659	0.995/0.751/0.274	0.996/0.828/0.385	N/A	0.996/0.847/**0.534**
SemiSupervised	0.999/0.838/**0.732**	0.999/0.778/0.291	0.999/0.870/**0.584**	N/A	1.000/0.883/0.358
SemiSupervised +NotCocoa	0.998/0.835/0.654	0.996/0.775/0.265	0.999/0.873/0.479	1.000/0.990/**0.695**	1.000/0.879/0.405
SemiSupervised +NotCocoa, DFLoss	1.000/0.955/0.631	1.000/0.900/**0.294**	1.000/0.926/0.401	1.000/0.983/0.619	1.000/0.957/0.513
SemiSupervised +NotCocoa, Focal Loss	1.000/0.864/0.678	1.000/0.810/0.174	1.000/0.718/0.421	1.000/1.000/0.613	1.000/0.774/0.449

Table 6.8: Percentage of difficult and unsure images relabelled during semi-supervised learning, and Grad-CAM score for each ResNet18 training variant. Grad-CAM score ranges from 0 to 1 and indicates how well the model focused on informative features. Bold text indicates values of particular interest.

Variant	% Difficult	% Unsure	Grad-CAM score
FullSupervised	N/A	N/A	0.200
SemiSupervised	24.36	**11.73**	0.288
SemiSupervised +NotCocoa	47.75	26.08	0.463
SemiSupervised +NotCocoa, DFLoss	**71.14**	35.60	**0.700**
SemiSupervised +NotCocoa, Focal Loss	61.69	40.55	**0.700**

6.3.2 Dynamic Focal Loss

Tables 6.3 to 6.5 show that applying DFLoss to PhytNet did not improve overall performance relative to the semi-supervised + non-cocoa baseline. Mean validation F1 fell from 0.755 to 0.722 and test F1 from 0.377 to 0.371. The main benefit was confined to the non-cocoa class, where test F1 increased from 0.492 to 0.554, while test WBD F1 dropped sharply from 0.176 to 0.102. The percentage of difficult and unsure images relabelled also remained high at 75.57% and 43.51%, respectively. DFLoss produced the best validation metrics for ResNet18, including a mean validation F1 of 0.949 and validation FPR F1 of 0.900, as shown in Tables 6.6 to 6.8. However, these gains did not transfer to the independent test set, where mean F1 was 0.510, below semi-supervised learning alone (0.549) and semi-supervised + non-cocoa (0.535). Even so, the DFLoss model retained the highest test precision (0.560), the highest test FPR F1 (0.294), a high test WBD F1 (0.513), the highest percentage of difficult images relabelled (71.14%), and a shared-highest Grad-CAM score (0.700).

For the capacity-comparison experiment, table 6.9 reports the optimised settings for PhytNet67k, PhytNet183k, and PhytNet332k. In the earlier optimisation sweep these three variants had similar validation F1 values (0.701, 0.693, and 0.699, respectively), with PhytNet67k selected as the baseline model for subsequent training-procedure comparisons.

Shown in fig. 6.4 (C) and Tables 6.10 to 6.12 are the results of an experiment to compare three PhytNet variants trained with cross-entropy loss and with DFLoss. The validation set still suggests that DFLoss can reduce overfitting for PhytNet183k, but the independent test set gives a more nuanced picture. PhytNet332k with cross-entropy loss achieved the best test accuracy (0.438) and mean F1 (0.439), while DFLoss improved PhytNet183k only modestly on test mean F1 (0.403 versus 0.385) and reduced PhytNet332k test mean F1 to 0.358. As such, the main effect of DFLoss appears to depend on model capacity rather than providing a uniform gain: it helps the 183k model slightly on the independent test set, but the stronger 332k cross-entropy model remains the best overall test performer.

Table 6.9: Comparison of parameters for different PhytNet configurations. All parameters were optimised for validation F1 in a WANDB optimisation sweep using a Bayesian process.

	PhytNet67k	**PhytNet183k**	**PhytNet332k**
beta1	0.9650	0.9671	0.9657
beta2	0.9816	0.9574	0.9908
conv. channels 1	79	126	104
conv. channels 2	107	91	109
conv. channels 3	93	89	110
image input size	415	371	350
kernel size 1	5	5	5
kernel size 2	1	1	7
kernel size 3	7	17	13
learning rate	0.000298	0.000967	0.000134
num. conv. blocks 1	2	2	1
num. conv. blocks 2	1	1	2
output channels	6	7	9
n parameters	67,302	183,227	331,852
GFLOPS	0.6049	0.8591	1.188

6.3.3 Focal Loss

The optimised value of FL gamma for PhytNet67k (1.0027) was close to 1, suggesting that the model performed best when the effect of FL was minimal. Nevertheless, Tables 6.3 to 6.5 show that, among the PhytNet variants in Tables 6.3 to 6.5, FL produced the strongest independent test result, with accuracy 0.388 and mean F1 0.384, and it also gave the highest test Healthy F1 (0.394). However, test FPR F1 remained low (0.192), the percentage of unsure images relabelled was the highest of any PhytNet variant (56.95%), and the Grad-CAM score remained low at 0.225.In contrast, the ResNet18 optimised FL gamma value was 2.885, which suggests that FL had a beneficial effect during the sweep. However, Tables 6.6 to 6.8 show that after semi-supervised training FL reduced mean validation F1 from 0.883 to 0.840 relative to the semi-supervised + non-cocoa baseline and reduced test F1 from 0.535 to 0.503. Although this variant retained a strong difficult-image relabelling rate (61.69%) and a shared-highest Grad-CAM score (0.700), it produced the weakest test FPR F1 of any ResNet18 variant (0.174).

6.3.4 PhytNet vs ResNet

Across the independent test set, ResNet18 remained clearly stronger overall than PhytNet. The best ResNet18 test result was obtained with semi-supervised learning alone, which reached accuracy 0.562 and mean F1 0.549 in Tables 6.6 to 6.8, whereas the best PhytNet test result in Tables 6.3 to 6.5 was obtained with semi-supervised + non-cocoa + FL at accuracy 0.388 and mean F1 0.384. ResNet18 also achieved higher best-in-class test scores for BPR (0.732), Healthy (0.584), and WBD (0.534), while PhytNet did not exceed ResNet18 on any overall test metric.

PhytNet nevertheless retained an advantage in the semi-supervised relabelling measures. Its best variants correctly relabelled about 77–78% of difficult images, compared with 71.14% for the best ResNet18 variant by this metric. This suggests that PhytNet remained more willing or able to incorporate difficult cocoa images during training, whereas ResNet18 transferred better to the independent test set. Taken together, the train, validation, and test results in Tables 6.3 to 6.8 show that validation performance alone would have overstated the benefits of the more aggressive ResNet18 variants, particularly the addition of the non-cocoa class and DFLoss.

As shown in fig. 6.5, both PhytNet and ResNet18 tended to perform well at attending to informative features. However, fig. 6.5 also shows cases where one or both models fail. For example, 1) both models are shown to correctly classify an image as BPR while attending to the bare soil (fig. 6.5 A). This is a textbook sign of overfitting to correlative features in the dataset. 2) PhytNet is shown to correctly classify an image as FPR while attending more to the healthy tissue than the lesion (fig. 6.5 C) while ResNet18, perhaps understandably, misclassifies this image as BPR while attending well to the necrotic tissue. 3) Both models correctly classify the image in fig. 6.5 (D), while PhytNet focuses its attention on irrelevant leaf litter in the background. This is a strong sign of dataset memorisation, a form of overfitting that will lead to complete failure upon deployment in the real world. And 4) both models are shown to attend well to the young nursery cocoa plants (fig. 6.5 F), yet both fail to classify these plants as healthy cocoa.

Runtime outcomes are reported in Table 6.13. ResNet18 required substantially more compute than PhytNet183k (GFLOPS: 5.24 vs 0.86) and achieved lower throughput (33 fewer FPS

on GPU and 15 fewer FPS on CPU). Table 6.13 also shows 4.96% higher GPU utilisation and 5% lower peak VRAM use for PhytNet183k relative to ResNet18.

Table 6.10: Performance metrics of PhytNet model variants with and without DFLoss applied. Train, validation, and test values are shown for loss, accuracy, precision, recall, and F1. Bold text indicates values of particular interest.

Variant	Metric	Train	Validation	Test
67k CE loss	Loss	0.422	0.692	2.604
	Accuracy	0.834	0.756	0.362
	Precision	0.838	0.760	0.419
	Recall	0.834	0.756	0.362
	F1 Score	0.834	0.755	0.377
183k CE loss	Loss	0.487	0.710	**2.123**
	Accuracy	0.805	0.749	0.363
	Precision	0.825	0.769	0.478
	Recall	0.805	0.749	0.363
	F1 Score	0.808	0.755	0.385
183k DFLoss	Loss	0.630	0.684	2.127
	Accuracy	0.790	0.790	0.394
	Precision	0.802	0.804	0.446
	Recall	0.790	0.790	0.394
	F1 Score	0.794	0.791	0.403
332k CE loss	Loss	0.250	0.770	2.510
	Accuracy	0.909	0.760	**0.438**
	Precision	0.914	0.775	**0.481**
	Recall	0.909	0.760	**0.438**
	F1 Score	0.908	0.760	**0.439**
332k DFLoss	Loss	0.447	0.589	2.773
	Accuracy	0.860	0.790	0.348
	Precision	0.874	0.811	0.453
	Recall	0.860	0.790	0.348
	F1 Score	0.860	0.794	0.358

Table 6.11: Per-class train (T), validation (V), and test (Te) F1 scores for PhytNet model variants with and without DFLoss. Bold text indicates values of particular interest.

Variant	BPR (T/V/Te)	FPR (T/V/Te)	Healthy (T/V/Te)	NotCocoa (T/V/Te)	WBD (T/V/Te)
67k CE loss	0.779/0.727/0.501	0.672/0.564/**0.186**	0.828/0.737/0.353	0.985/0.941/0.492	0.806/0.682/0.176
183k CE loss	0.766/0.772/0.496	0.652/0.579/0.239	0.774/0.706/0.309	0.974/0.915/0.573	0.779/0.714/0.158
183k DFLoss	0.583/0.773/0.533	0.718/0.720/0.228	0.708/0.745/**0.335**	0.982/0.923/**0.584**	0.824/0.714/0.163
332k CE loss	**0.852**/0.748/**0.595**	0.823/0.598/0.189	0.915/0.731/0.292	0.984/0.925/0.531	0.918/0.691/**0.504**
332k DFLoss	0.867/0.765/0.503	0.718/0.649/0.221	0.773/0.776/0.160	0.985/0.982/0.582	0.909/0.609/0.244

Table 6.12: Percentage of difficult and unsure images relabelled during semi-supervised learning, and Grad-CAM score for each PhytNet model variant with and without DFLoss. Grad-CAM score ranges from 0 to 1 and indicates how well the model focused on informative features. Bold text indicates values of particular interest.

Variant	% Difficult	% Unsure	Grad-CAM score
67k CE loss	77.93	40.66	0.588
183k CE loss	77.23	**47.27**	0.513
183k DFLoss	65.40	46.24	0.463
332k CE loss	**83.70**	39.84	**0.638**
332k DFLoss	67.08	23.00	0.425

Table 6.13: Runtime performance metrics for two neural network models. The GPU used was an Nvidia H100 80GB and the CPU was an AMD EPYC 7643 48-core processor. Batch size: 100

Model	n params	GFLOPS	FPS (GPU)	FPS (CPU)	GPU util. (%)	Peak VRAM. usage (MB)
ResNet18	11,689,512	5.24	963	85	35.65	280.07
PhytNet183K	183,227	0.86	996	100	40.61	266.46

6.4 Discussion

In this chapter, we explored various training strategies and model enhancements to improve the performance of convolutional neural networks. We focus on ResNet18 and PhytNet in the detection of diseases in cocoa plants while incorporating a discussion on runtime performance metrics. The overarching findings reveal significant advancements in model accuracy, generalisation capabilities, and interpretability through the incorporation of semi-supervised learning, the addition of a non-cocoa class, and the implementation of specialised loss functions, FL and DFLoss. Additionally, the use of Grad-CAM for qualitative assessment provided deeper insights into model behaviour, offering a more nuanced understanding of where these models focus their attention and how this relates to their predictions.

6.4.1 Efficacy of Semi-Supervised Learning

The semi-supervised learning approach used here markedly improved all performance metrics for both PhytNet and ResNet18. Many previous studies have used semi-supervised learning to incorporate both labelled and unlabeled data into the training process to effectively enhance the model's learning capability, particularly in scenarios where labelled data is unavailable or expensive to obtain [152]. However, here we show that it can be used to allow models to learn from correctly labelled images with symptoms that were difficult to detect with the human eye or invisible to humans, without forcing overfitting. While this form of semi-supervised learning still necessitates the laborious and costly task of data labelling, we show here that this effort is well spent in training models that generalise better to the real world than those trained with full supervision. This effect is apparent in the qualitative Grad-CAM scores, which showed a 37.5% and 87.5% increase for PhytNet and ResNet18 re-

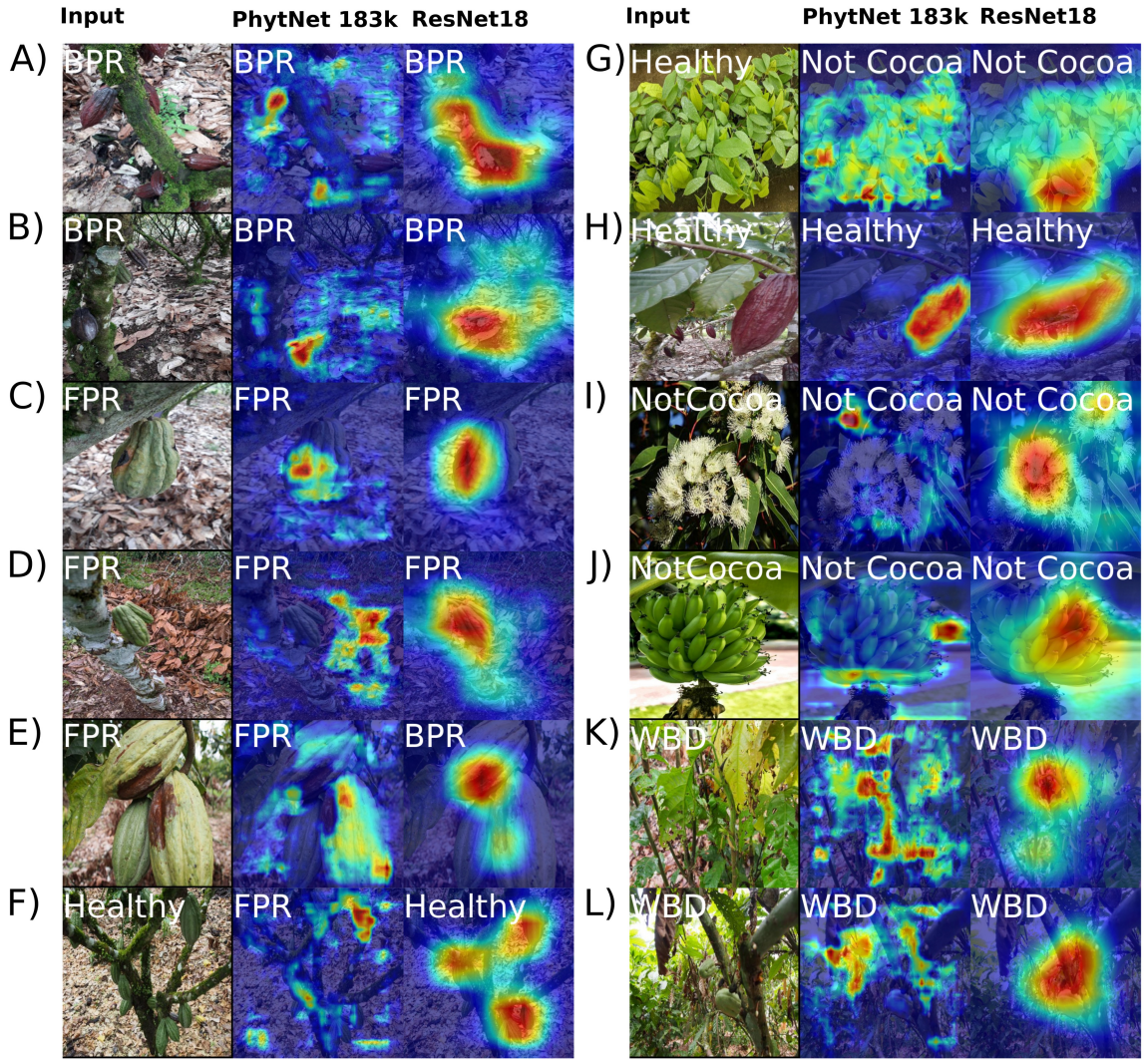


Figure 6.5: Grad-CAM class activation maps comparing the best performing PhytNet183k and ResNet18 models. PhytNet183k was trained with semi-supervised learning, the additional cocoa class, and cross-entropy loss. ResNet18 was trained with semi-supervised learning, the additional cocoa class, and DFLoss.

spectively with this approach. Crucially, in PhytNet67k semi-supervised learning increased the FPR validation score by 20.9%. This improvement to the FPR F1 score is of critical importance to the application of these models in the field as FPR poses a significant risk if it spreads to Brazilian or African crops [57, 63] and it is the most difficult of the three diseases to detect [241].

6.4.2 Inclusion of the Non-Cocoa Class

With inclusion of the non-cocoa class we aimed to prevent overfitting to irrelevant features by exposing the model to a wider variety of features that it is likely to encounter in the real world. A concerted effort was made during data collection to include features that would help prevent overfitting. The inclusion of the non-cocoa class extended this effort to include images of a great variety of plants and disease symptoms. Features of such images are distinct from but could be mistaken for those relevant to the detection of cocoa disease. For example, *Phytophthora spp.*, which causes BPR, infects dozens of other crop species [124], many of which are included here in the non-cocoa class. The inclusion of this class improved all performance metrics for both models and did so to a greater degree than any other additional method employed here. Moreover the dramatic increase in both the number of difficult and unsure images relabelled and the improved Grad-CAM scores, coupled with a reduced disparity in training vs validation values shows that the inclusion of this class had a marked beneficial effect in the reduction of overfitting.

6.4.3 Focal Loss and Dynamic Focal Loss

While standard FL was developed to address the problem of massively imbalanced classes in binary data, it is also intended to help focus model attention more on observations that the model struggles to classify. Similarly, DFLoss was developed and implemented here to address the challenge of learning from difficult or less frequent observations in a multi-class scenario and should also help with cases of class imbalance. This latter feature was not tested here as the present data are well-balanced but will be tested in future work. The varied results observed in testing DFLoss across different models here indicate that its effectiveness may be model-specific. Tables 6.3 to 6.8 show that DFLoss helps ResNet18 on validation but not on the independent test set, whereas PhytNet benefits only marginally and inconsistently. This behaviour is likely due to the fact that PhytNet is designed explicitly to be appropriately parameterised for a given dataset. So here, after running the PhytNet optimisation sweep with cross-entropy loss, PhytNet lacked the spare capacity to make use of DFLoss. By contrast, the present results show that, without DFLoss, ResNet18 is clearly over-parameterised to this dataset and so it had the spare capacity to do well with DFLoss.

To test this explanation in future work, we will evaluate PhytNet's performance using DFLoss prior to the optimisation sweep.

If it is not possible to avoid the use of an over-parameterised model, DFLoss offers potential benefits in improving model fit by utilising the spare model capacity. Potential reasons why DFLoss would decrease overfitting in ResNet18 include: 1) that it is acting as a form of regularisation by preventing the model from overly focusing on the more frequent or easily recognisable features, or 2) given that training accuracy was 99.9% without DFLoss and so the model could hardly fit more to the training data, DFLoss might have caused the model to learn the genuine patterns of some images before it learned too many false patterns.

FL appears to increase the validation metrics of PhytNet slightly but it did not affect the percentage of difficult images relabelled, it greatly increased the percentage of unsure images relabelled and it greatly reduced the Grad-CAM score. Based on these results FL only increased overfitting in PhytNet without any improvement in its ability to classify images. By contrast, FL appears to decrease overfitting in ResNet18, though less so than DFLoss. FL led to a marked improvement in the percentage of difficult images relabelled and the joint highest Grad-CAM score of any model variant, but Tables 6.6 to 6.8 show that this did not translate into better test-set F1.

As with model pruning, rather than using tools like FL to combat overfitting retrospectively, it should be preferable to begin with a model that is appropriately scaled to the dataset at hand [242]. While a great deal of research effort has been applied to model pruning and quantisation without loss of performance, some loss appears inevitable [243, 211]. Similarly, the results presented here for FL further vindicate the use of appropriately scaled models and the avoidance of hacking model behaviour in convoluted ways. However, the effectiveness of DFLoss with ResNet18 appears to contradict this narrative to some degree. To investigate this matter further, we should develop new means of detecting and explaining overfitting in ResNet18 with DFLoss. This is because the very low training loss suggests it is still overfitting, even if all other metrics show it is also learning genuine patterns in the data very well. Additional work is also required to improve the performance of PhytNet which shows great potential in its superior ability to classify difficult images over ResNet18 with far fewer parameters and with greater computation efficiency.

6.4.4 Grad-CAM Analysis

The use of the Grad-CAM score for quantifying the inherently qualitative analysis of reviewing class activation maps adds a novel dimension to the present evaluation of model performance. While the subjective nature of such analyses can pose difficulties in objectivity and reproducibility, when paired with other rigorous quantitative analyses, they can be highly informative. Visual inspection of data and results is vital in data analysis but in huge datasets typical of analyses with deep neural networks, objective comparisons become difficult. The simple scoring system employed here worked well in allowing for the subjective analysis of model attention through class activation maps and a meaningful comparison of models. For example, when paired with the decrease in the percentage of difficult images relabelled, the decreased Grad-CAM score allows for the differentiation between PhytNet183K with and without DFLoss in table 6.12.

6.4.5 Model Selection

Considering the wealth of performance data presented here, the selection of the 'best' model is far from simple, though the addition of the independent test set strongly suggests that ResNet is the superior model. There are three main criteria by which we will judge model performance; 1) summary statistics such as accuracy and F1 score, 2) model attention and ability to generalise, and 3) runtime speed and compute requirements.

6.4.5.1 Summary Statistics

While metrics like F1 and loss are highly informative, model choice is not a simple matter of choosing the highest validation F1 and lowest loss. Rather, we must consider these metrics in context as doing so can provide meaningful insights into the fit of the model and its behaviour. Tables 6.3 to 6.8 make that point clearly: the best validation scores often come from the more complex variants, but the best independent test scores do not. For example, when trained with cross entropy loss ResNet18 scored an impressive 99.9% training F1 and 88.3% validation F1. However with FL added, the same model scored 4% lower validation F1 but relabelled 14% more difficult images and 14% more unsure images, *i.e.* it performed markedly

worse on easy images but much better with difficult and unsure images. Additionally, both PhytNet and ResNet18 are shown to focus on bare soil at the base of a tree while correctly classifying the image as BPR.

Tables 6.3 to 6.5 show that PhytNet183K with cross-entropy loss scored one of the highest validation F1 values (75.5%) with the second lowest training F1 (80.8%), but the best PhytNet test result was instead the semi-supervised + non-cocoa + focal-loss variant. This, in conjunction with the modest Grad-CAM score, suggests that PhytNet183k fit best to the training distribution without obvious overfitting, but it did not produce the strongest independent test performance. Its Grad-CAM score was exceeded by as much as 15% by other PhytNet variants, again highlighting the complexity of model choice. Additionally, the metrics of PhytNet are generally dwarfed by those of the best ResNet18 test variant, though this is not true of any of the other ResNet18 variants, which had quite variable performance metrics and showed many signs of overfitting.

6.4.5.2 Model Attention and Ability to Generalise

In Tables 6.3 to 6.8, we see that only one of the PhytNet variants (67k: 0.588, 183k: 0.513 & 332k: 0.663) performed comparably to the best ResNet18 variants (0.7). However, we also see that an improved Grad-CAM score does not directly correlate with reduced overfitting, as we might expect. The Grad-CAM scores and fig. 6.5 show that, despite ResNet18 showing evidence of overfitting in its loss values, it does well in many cases at focusing its attention on relevant features. This suggests that, while ResNet18 may have overfit to the data, it also learned many relevant features. However, PhytNet appears to have learned many of these same relevant features with two to three orders of magnitude fewer trainable parameters. While PhytNet332k had the second-greatest Grad-CAM score, it also showed signs of overfitting by large disparities between training and validation metrics. However, this disparity was reduced by DFLoss. Additionally, despite being 45% larger than PhytNet183k, PhytNet332k only scored 0.3-0.5% greater validation F1 than PhytNet67k. For these reasons, PhytNet183k is likely the best of the PhytNet variants for efficiency, but the best independent test performance comes from ResNet18 semi-supervised learning alone. One might still choose PhytNet332k due to the high percentage of difficult images that it relabelled (84%)

or its high grad-CAM scores (64%) with cross-entropy loss.

6.4.5.3 Runtime Speed and Compute Requirements

While there are many metrics by which we can compare the runtime speed and compute requirements of two models, by far the most important for real-world deployment are frames per second (FPS) and peak VRAM utilisation. While the difference in VRAM utilisation between the PhytNet183k and ResNet18 is negligible, the difference in FPS on a CPU and GPU is more pronounced. Scaled up, these differences amount to an extra 118,800 images per hour with PhytNet183k over ResNet18 on a GPU. As such, if the chosen model is to be run on low-powered hardware or if large scale is required then PhytNet183k may be preferable. This, coupled with PhytNet's better relabelling behaviour and relatively high Grad-CAM attention scores, suggests that PhytNet has the potential to out-compete ResNet18 for this dataset on efficiency grounds. However, Tables 6.6 to 6.8 show that the best independent test performance is still achieved by ResNet18 semi-supervised learning alone.

6.5 Summary

Through semi-supervised learning, specialised loss functions, and the inclusion of a non-cocoa class, we achieved improvements in model accuracy, generalisation, and interpretability. The split results in Tables 6.3 to 6.8 show that ResNet18 achieved the strongest independent test performance, while PhytNet retained the advantage in relabelling difficult images and computational efficiency. The Grad-CAM analysis added a valuable qualitative dimension, revealing where models focused their attention and providing insights into their decision-making processes.

While we selected ResNet18 and PhytNet for their balance of performance and computational efficiency, this study did not include the latest transformer-based models such as Real-time Detection Transformer (DETR) [129], which, although significantly larger in number of parameters, offer fast runtime speeds and could provide a fruitful area for future work. Additionally, the reliance on fully labelled data for supervised learning remains a limitation, as it is both labour-intensive and costly. Further research should explore methods to reduce

this dependency and investigate the generalisability of these models across more diverse datasets.

By addressing these challenges, the continued development of efficient and robust models like PhytNet can play a critical role in real-world disease monitoring and intervention, enabling scalable solutions for sustainable food production.

Chapter 7

Future Work

Several directions remain to extend and translate this work. These priorities consolidate the future-work needs raised across the thesis, including the spectroscopy limitations in Chapter 4, the architectural observations in Chapter 5, and the advanced-training and deployment questions in Chapter 6.

1. **Field-Deployment & User Studies.** Integrate the trained PhytNet or ResNet18 models into handheld or drone-mounted devices, undertaking pilot trials with cocoa smallholders to assess usability, inference latency, and economic impact on spray-timing and yield preservation. This would require collaboration with local partners to design robust hardware interfaces, establish wireless or offline data pipelines, and develop intuitive user interfaces suitable for field technicians.

2. **Stronger Non-Visible and Multi-Modal Evaluation.** Fuse visual sensing with complementary data streams such as environmental sensing, molecular assays, spectroscopy, and farmer-reported metadata to build a more holistic disease-forecasting system. This should include synchronising multimodal datasets via Global Positioning System (GPS) timestamps, adapting the model architecture to handle heterogeneous inputs, and validating predictive accuracy across environmental gradients. For spectroscopy in particular, the immediate priority is to collect more genuinely independent batches or days of acquisition, control cultivar, pod temperature, disease stage, and ambient conditions more tightly, and explicitly test batch-harmonisation or domain-

adaptation methods before drawing strong conclusions about wavelength importance.

3. **Label Efficiency, Continual Learning, and Weak Supervision.** Develop training pipelines that reduce dependence on fully labelled data and allow models to adapt continuously to new cultivars, emerging disease strains, or shifting ecological contexts with minimal annotation cost. This should include active learning, uncertainty-guided sample review, semi-supervised updating, and on-device or low-cost continual-learning strategies that avoid catastrophic forgetting. A key translational question is whether the benefits observed from relabelling and non-cocoa negatives in Chapter 6 can be extended into a sustainable human-in-the-loop workflow.

4. **Architecture Validation and Transfer.** Test whether PhytNet's current advantages remain stable across more complex datasets, additional crops, and transfer-learning settings. This includes evaluating whether PhytNet is dataset-dependent, whether its compactness leads to underfitting in harder settings, and whether plant-pathology-specific pre-training is more effective than generic transfer from datasets such as ImageNet. The unusual result that seven or eight output nodes outperformed four in Chapter 5 should also be investigated directly to determine whether the extra capacity acts as regularisation, latent subclass separation, or a mechanism for handling uncertain or mixed symptom states.

5. **Loss Functions, Model Capacity, and Generalisation.** Extend the analysis of dynamic focal loss (DFLoss) by testing it in settings that were left open in the present thesis. Two immediate experiments are needed: evaluating DFLoss on genuinely imbalanced cocoa data, and testing PhytNet with DFLoss before the architecture optimisation sweep to determine whether the mixed results seen in Chapter 6 were caused by limited spare model capacity. More broadly, future work should compare advanced loss functions against explicit model-capacity control, calibration methods, and harder independent test sets to identify when these procedures improve true generalisation rather than only validation performance.

6. **Generalisation Across Crops, Devices, and Modern Model Families.** Extend the framework to additional high-value crops and to broader acquisition conditions, while also testing stronger modern baselines. This includes validating performance

across countries, canopy structures, cultivars, disease morphologies, and camera hardware, as well as comparing PhytNet and ResNet18 against newer efficient transformer-based or detector-style models where deployment cost remains realistic. In parallel, deployment-oriented compression methods such as pruning, quantisation-aware training, and related optimisation strategies should be revisited under plant-pathology constraints rather than assumed to transfer directly from generic benchmarks.

Chapter 8

Conclusions

> "There is more pollen on my face than when I began. New whys have fallen on top of old whys. There is a bigger pile to leap into, and it smells just as mysterious as it did at the start."
>
> *Entangled life, Merlin Sheldrake*

8.1 Revisiting the Research Questions and Hypotheses

The thesis research questions and hypotheses were stated in Section 1.10. This section revisits them in light of the evidence presented across Chapters 4 to 6.

RQ1/H1 asked whether a tailored lightweight convolutional architecture could match or exceed the practical performance of larger off-the-shelf models while reducing computational cost. The evidence presented in Chapter 5 supports this hypothesis in substantial part. PhytNet was consistently competitive with larger architectures and achieved this with far fewer parameters and lower compute requirements. Although ResNet18 remained a strong benchmark and, on the independent test set in Chapter 6, ultimately achieved the strongest overall test performance under its best training procedure, the thesis still provides clear evidence that a tailored lightweight architecture can deliver a superior efficiency-to-performance

trade-off. PhytNet therefore answers RQ1 positively in practical terms, even if the answer is not that lightweight models dominate larger ones on every metric.

RQ2/H2 asked whether non-visible signals, especially infrared and spectroscopy-derived signatures, improve disease discrimination and early symptom detection beyond visible-spectrum imaging alone. The evidence here is mixed and does not justify a strong affirmative statement. The spectroscopy chapter showed that the available field spectra were highly batch-sensitive and did not support robust generalisation under leave-one-batch-out validation. At the same time, the imaging experiments in Chapter 5 showed a modest but meaningful infrared (IR) advantage for some tasks, especially for frosty pod rot (FPR). The resulting answer to RQ2 is therefore cautious: non-visible sensing may contain useful discriminatory information, but this thesis does not provide sufficient evidence to claim that spectroscopy or other non-visible modalities reliably outperform visible-spectrum imaging under realistic independent evaluation. H2 is therefore only partially supported.

RQ3/H3 asked whether advanced training procedures, including semi-supervised relabelling, dynamic focal loss, and the addition of a non-cocoa class, improve model fit to difficult real-world cases and reduce overfitting. The evidence from Chapter 6 supports this hypothesis, but with important qualification. Semi-supervised learning and the non-cocoa class clearly improved model behaviour, especially by improving difficult-case handling, reducing reliance on spurious cues, and increasing performance on both validation and independent test data for several model variants. By contrast, dynamic focal loss was not uniformly beneficial: it helped some configurations, particularly over-parameterised ResNet18 and certain PhytNet capacity settings, but did not provide a consistent improvement across all models. The answer to RQ3 is therefore yes for the broader training strategy, but no for any claim that all advanced loss-based procedures help equally. H3 is supported in general structure, not in every component equally.

RQ4/H4 asked whether model interpretability and deployment suitability can be improved concurrently, such that stronger attention to biologically relevant features is achieved without compromising runtime feasibility for low-resource use. The thesis provides evidence that this is achievable, though again with trade-offs. Across Chapters 5 and 6, gradient-weighted class activation mapping (Grad-CAM) analysis showed which of the stronger-performing

models focused on relevant lesions, pod morphology, and symptomatic tissue rather than only on background confounders. Runtime profiling in Chapter 6 further showed that PhytNet183k retained a clear compute advantage over ResNet18 while remaining competitive in classification performance. The answer to RQ4 is therefore affirmative: interpretability and deployment feasibility can be improved together, although the single best model by accuracy is not identical to the single best model by efficiency.

RQ0/H0, the overarching research question and hypothesis from Section 1.10, asked whether biologically informed and computationally efficient computer vision systems can provide accurate, generalisable, and interpretable detection of major cocoa diseases under realistic field conditions, and whether a data-centric pipeline combining tailored model design, biologically informed data acquisition, and robust training procedures would outperform generic workflows for real-world cocoa disease detection. Taken together, the evidence in this thesis supports that overarching claim. The strongest support comes not from any single model or sensor modality, but from the pipeline as a whole: careful dataset design, lightweight architecture development, leakage-aware evaluation, advanced training procedures, interpretability checks, and independent test-set validation. The thesis therefore concludes that biologically informed and computationally efficient computer vision systems can indeed provide a credible basis for real-world cocoa disease detection, provided that claims of success are grounded in genuinely independent evaluation rather than any one metric or set of naive metrics alone.

8.2 Limitations and Assumptions

The findings of this thesis should be interpreted within several clear limitations and assumptions. First, the work is centred on cocoa disease detection in Ecuadorian field conditions, with datasets collected from a limited number of sites, seasons, devices, and acquisition workflows. Although the independent Fairfield Vision test set provides a materially stronger assessment of generalisation than validation data alone, it still reflects broadly similar regional conditions rather than a fully global deployment setting. The conclusions are therefore strongest for field-realistic cocoa monitoring in related contexts, and weaker for claims about transfer across countries, cultivars, climates, or imaging hardware without further validation.

Second, the thesis assumes that the available labels are sufficiently reliable to support supervised and semi-supervised training, while also recognising that agricultural image labels are inherently imperfect. Some classes, especially early, ambiguous, or mixed symptom cases, are difficult even for experts to separate cleanly from images alone. The semi-supervised relabelling framework improved practical behaviour in several experiments, but it does not remove the possibility that some residual label noise, latent class overlap, or hidden confounding factors remain in the datasets. The reported metrics should therefore be read as strong comparative evidence within the constructed benchmarks, not as proof that all images were labelled without error.

Third, the work assumes that classification performance, attention-map inspection, and runtime profiling together provide a useful basis for judging deployment suitability. This is defensible for the technical scope of the thesis, but it is not equivalent to a complete translational evaluation. Grad-CAM gives only a partial and approximate view of model reasoning, and runtime measurements on the tested hardware do not by themselves guarantee performance on every embedded or mobile platform. In the same way, good image-level classification does not fully address downstream requirements such as disease severity estimation, farm-level decision support, user trust, maintenance costs, or integration with extension workflows.

Fourth, the non-visible sensing results depend on assumptions about acquisition stability that were only partly satisfied in practice. The spectroscopy experiments showed strong batch sensitivity, which limits confidence in broader claims about spectral superiority. The infrared results were more encouraging, but they were still tested within a bounded experimental setup rather than across a wide range of operational environments. The thesis therefore supports cautious optimism for biologically informed sensing, but not a general claim that adding sensors will automatically improve disease detection performance.

Finally, the thesis assumes that a data-centric and computationally efficient approach is an appropriate optimisation target for plant pathology artificial intelligence (AI). That assumption is justified by the low-resource and field-deployment constraints motivating the work, but it also means the research does not exhaust the space of possible high-capacity, multimodal, segmentation-first, or transformer-first solutions. The conclusions should therefore

be understood as evidence for a practical and defensible pathway to real-world cocoa disease detection, rather than as a claim that the final models or methods presented here are definitive endpoints.

8.3 Closing Remarks

Over the next five to ten years, plant-pathology AI is likely to move away from small demonstrations of high validation accuracy and towards integrated decision-support systems that are tested under genuine agronomic conditions. The strongest work in the field will likely combine image analysis with temporal, environmental, and operational data, evaluate models on truly independent and geographically broader test sets, and place much greater emphasis on calibration, uncertainty, and failure-mode analysis. In that sense, the direction of travel is away from isolated benchmark performance and towards robust systems evidence.

At the same time, model development is likely to become more heterogeneous. Compact architectures such as PhytNet will remain important where cost, power, and deployment simplicity matter, while larger transformer-based or multimodal models may become increasingly practical as hardware availability improves. The key question will probably no longer be whether one model family dominates all others, but which level of model complexity is justified for a given biological task, operating constraint, and evidence standard. This thesis suggests that efficiency, interpretability, and leakage-aware evaluation should remain central in that comparison.

The biological side of the field is also likely to mature. Future systems will need to account more explicitly for symptom progression, mixed infections, cultivar effects, phenology, and the distinction between visible damage and underlying pathogen presence. As datasets improve, the most useful advances may come not simply from larger models, but from tighter integration between plant pathology knowledge, sensor design, and machine learning methodology. If that happens, the field can move beyond retrospective disease recognition towards earlier detection, better targeting of interventions, and more credible support for growers making real management decisions.

Taken together, the next decade is likely to reward research that is technically strong but also

biologically grounded, operationally realistic, and honest and quantitative about uncertainty. That is the direction in which this thesis places its contribution: not as a final solution, but as part of the transition from promising computer-vision prototypes to dependable crop-health tools.

List of Acronyms

AI	artificial intelligence
ML	machine learning
CV	computer vision
CNN	convolutional neural network
DNN	deep neural network
GPU	graphics processing unit
CPU	central processing unit
RGB	red, green, and blue
IR	infrared
UV	ultraviolet
UVA	ultraviolet A
NIR	near-infrared
HSI	hyperspectral imaging
VAE	variational autoencoder
NVAE	Nouveau variational autoencoder
SVM	support vector machine
PCA	principal component analysis

t-SNE	t-distributed stochastic neighbor embedding
UMAP	uniform manifold approximation and projection
GAN	generative adversarial network
YOLO	You Only Look Once
DETR	Detection Transformer
RT-DETR	real-time Detection Transformer
DINO-DETR	DETR with improved de-noising anchor boxes
R-CNN	region-based convolutional neural network
R-FCN	region-based fully convolutional network
Faster R-CNN	Faster region-based convolutional neural network
Mask R-CNN	Mask region-based convolutional neural network
COCO	Common Objects in Context
Grad-CAM	gradient-weighted class activation mapping
BPR	black pod rot
FPR	frosty pod rot
WBD	witches' broom disease
NPQt	non-photochemical quenching

Phi2	photosystem II quantum yield
F1	F1 score
AUC	area under the curve
mAP	mean average precision
FPS	frames per second
GFLOPS	giga floating-point operations per second
VRAM	video random-access memory
BN	batch normalization
IN	image normalization
LN	layer normalization
ReLU	rectified linear unit
GELU	Gaussian error linear unit
SE	squeeze-excitation
LAMP	loop-mediated isothermal amplification
PCR	polymerase chain reaction
qPCR	quantitative polymerase chain reaction
RT-qPCR	real-time quantitative polymerase chain reaction
DNA	deoxyribonucleic acid
RNA	ribonucleic acid
ITS	internal transcribed spacer

ONNX	Open Neural Network Exchange
API	application programming interface
FAIGB	forestry and arable images from Google and Bing
DFLoss	dynamic focal loss
FL	focal loss
CE	cross-entropy loss
GPS	Global Positioning System
INIAP	Instituto Nacional de Investigaciones Agropecuarias
ICQC	International Cocoa Quarantine Centre
EPSRC	Engineering and Physical Sciences Research Council
wandb	Weights and Biases
UK	United Kingdom
USA	United States of America

Bibliography

[1] D. I. Patrício and R. Rieder, "Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review," *Computers and Electronics in Agriculture*, vol. 153, pp. 69–81, Oct. 2018.

[2] Z. Wu, Y. Chen, B. Zhao, X. Kang, and Y. Ding, "Review of Weed Detection Methods Based on Computer Vision," *Sensors*, vol. 21, p. 3647, Jan. 2021.

[3] P. Ramos-Giraldo, C. Reberg-Horton, A. M. Locke, S. Mirsky, and E. Lobaton, "Drought Stress Detection Using Low-Cost Computer Vision Systems and Machine Learning Techniques," *IT Professional*, vol. 22, pp. 27–29, May 2020.

[4] R. S. Sarkate, N. V. Kalyankar, and P. B. Khanale, "Application of computer vision and color image segmentation for yield prediction precision," in *2013 International Conference on Information Systems and Computer Networks*, pp. 9–13, Mar. 2013.

[5] M. K. Tripathi and D. D. Maktedar, "A role of computer vision in fruits and vegetables among various horticulture products of agriculture fields: A survey," *Information Processing in Agriculture*, vol. 7, pp. 183–203, June 2020.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

[7] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," Apr. 2017.

[8] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6848–6856, 2018.

[9] C. A. Mack, "Fifty Years of Moore's Law," *IEEE Transactions on Semiconductor Manufacturing*, vol. 24, pp. 202–207, May 2011.

[10] S. Fuentes, G. Chacon, D. D. Torrico, A. Zarate, and C. Gonzalez Viejo, "Spatial Variability of Aroma Profiles of Cocoa Trees Obtained through Computer Vision and Machine Learning Modelling: A Cover Photography and High Spatial Remote Sensing Application," *Sensors*, vol. 19, p. 3054, Jan. 2019.

[11] P. Parra, T. Negrete, J. Llaguno, and N. Vega, "Computer Vision Techniques Applied in the Estimation of the Cocoa Beans Fermentation Grade," in *2018 IEEE ANDESCON*, pp. 1–10, Aug. 2018.

[12] M. M. Oliveira, B. V. Cerqueira, S. Barbon, and D. F. Barbin, "Classification of fermented cocoa beans (cut test) using computer vision," *Journal of Food Composition and Analysis*, vol. 97, p. 103771, Apr. 2021.

[13] K. Mite-Baidal, E. Solís-Avilés, T. Martínez-Carriel, A. Marcillo-Plaza, E. Cruz-Ibarra, and W. Baque-Bustamante, "Analysis of Computer Vision Algorithms to Determine the Quality of Fermented Cocoa (Theobroma Cacao): Systematic Literature Review," in *ICT for Agriculture and Environment* (R. Valencia-García, G. Alcaraz-Mármol, J. del Cioppo-Morstadt, N. Vera-Lucio, and M. Bucaram-Leverone, eds.), Advances in Intelligent Systems and Computing, (Cham), pp. 79–87, Springer International Publishing, 2019.

[14] J. Su, C. Liu, M. Coombes, X. Hu, C. Wang, X. Xu, Q. Li, L. Guo, and W.-H. Chen, "Wheat yellow rust monitoring by learning from multispectral UAV aerial imagery," *Computers and Electronics in Agriculture*, vol. 155, pp. 157–166, Dec. 2018.

[15] M. T. Chiu, X. Xu, Y. Wei, Z. Huang, A. G. Schwing, R. Brunner, H. Khachatrian, H. Karapetyan, I. Dozier, G. Rose, D. Wilson, A. Tudor, N. Hovakimyan, T. S. Huang, and H. Shi, "Agriculture-Vision: A Large Aerial Image Database for Agricultural Pattern Analysis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2828–2838, 2020.

[16] S. Nagarajan, G. Seibold, J. Kranza, E. E. Saari, and L. M. Joshi, "Monitoring Wheat Rust Epidemics With the Landsat-2 Satellite," *Phytopathology*, vol. 74, no. 5, p. 585, 1984.

[17] K. Grosch, "John Deere – Bringing AI to Agriculture," 2018.

[18] A. C. Maddison, G. Macias, C. Moreira, R. Arias, and R. Neira, "Cocoa production in Ecuador in relation to dry-season escape from pod rot caused by Crinipellis perniciosa and Moniliophthora roreri," *Plant Pathology*, vol. 44, no. 6, pp. 982–998, 1995.

[19] J.-P. Marelli, D. I. Guest, B. A. Bailey, H. C. Evans, J. K. Brown, M. Junaid, R. W. Barreto, D. O. Lisboa, and A. S. Puig, "Chocolate Under Threat from Old and New Cacao Diseases," *Phytopathology®*, vol. 109, pp. 1331–1343, Aug. 2019.

[20] W. Phillips-Mora and M. J. Wilkinson, "Frosty Pod of Cacao: A Disease with a Limited Geographic Range but Unlimited Potential for Damage," *Phytopathology®*, vol. 97, pp. 1644–1647, Dec. 2007.

[21] L. W. Meinhardt, J. Rincones, B. A. Bailey, M. C. Aime, G. W. Griffith, D. Zhang, and G. a. G. Pereira, "Moniliophthora perniciosa, the causal agent of witches' broom disease of cacao: What's new from this old foe?," *Molecular Plant Pathology*, vol. 9, no. 5, pp. 577–588, 2008.

[22] R. A. Rice and R. Greenberg, "Cacao Cultivation and the Conservation of Biological Diversity," *AMBIO: A Journal of the Human Environment*, vol. 29, pp. 167–173, May 2000.

[23] D. T. Kuok Ho and P. S. Yap, *A Systematic Review of Slash-and-Burn Agriculture as an Obstacle to Future-Proofing Climate Change*. TIIKM Publishing, July 2020.

[24] Y. Malhi, J. T. Roberts, R. A. Betts, T. J. Killeen, W. Li, and C. A. Nobre, "Climate Change, Deforestation, and the Fate of the Amazon," *Science*, vol. 319, pp. 169–172, Jan. 2008.

[25] Y. Cohen and M. D. Coffey, "Systemic Fungicides and the Control of Oomycetes," *Annual Review of Phytopathology*, vol. 24, no. 1, pp. 311–338, 1986.

[26] Department of Health, Victoria, Australia, "Methyl bromide use in victoria community factsheet," Oct. 2014. Available at: https://www.health.vic.gov.au/publications/methyl-bromide-use-in-victoria-community-factsheet (Accessed: 26 June 2023).

[27] Statista Research Department, "Cocoa bean production worldwide by region 2019/2020." https://www.statista.com/statistics/263855/cocoa-bean-production-worldwide-by-region/, 2020. Accessed: 2022-05-16.

[28] G. O. Essegbey and E. Ofori-Gyamfi, "Ghana Cocoa Industry—An Analysis from the Innovation System Perspective," *Technology and Investment*, vol. 3, pp. 276–286, Nov. 2012.

[29] G. Ton, G. Hagelaars, A. Laven, and S. Vellema, "Chain Governance, Sector Policies and Economic Sustainability in Cocoa: A Comparative Analysis of Ghana, Côte D'Ivoire, and Ecuador," SSRN Scholarly Paper 1609686, Social Science Research Network, Rochester, NY, Jan. 2008.

[30] Statista Research Department, "Cocoa bean production worldwide by country, 2019/2020." https://www.statista.com/statistics/263855/cocoa-bean-production-worldwide-by-region/, 2020. Accessed on 16 May 2022.

[31] Reuters, "Ecuador set to become world's no. 2 cocoa grower, industry head says." https://www.investing.com/news/commodities-news/ecuador-set-to-become-worlds-no-2-cocoa-grower-industry-head-says-4248525, Sept. 2025. Reported by May Angel for Reuters; accessed 5 April 2026.

[32] G. W. Griffith, J. Nicholson, A. Nenninger, R. N. Birch, and J. N. Hedger, "Witches' brooms and frosty pods: Two major pathogens of cacao," *New Zealand Journal of Botany*, vol. 41, pp. 423–435, Sept. 2003.

[33] R. S. Middendorp, O. Boever, X. Rueda, and E. F. Lambin, "Improving smallholder livelihoods and ecosystems through direct trade relations: High-quality cocoa producers in Ecuador," *Business Strategy & Development*, vol. 3, no. 2, pp. 165–184, 2020.

[34] K. Alger and M. Caldas, "The declining cocoa economy and the Atlantic Forest of Southern Bahia, Brazil: Conservation attitudes of cocoa planters," *Environmentalist*, vol. 14, pp. 107–119, June 1994.

[35] M. M. Caldas and S. Perz, ""Agro-terrorism? The causes and consequences of the appearance of witch's broom disease in cocoa plantations of southern Bahia, Brazil"," *Geoforum*, vol. 47, pp. 147–157, June 2013.

[36] P. Aikpokpodion and J. Motamayor, "Deploying genomics and breeding to improve cocoa productivity," *Plant Breeding*, vol. 138, no. 4, pp. 1–14, 2019.

[37] J. Kieckbusch and S. Beck, "Plant diversity effects on coca agroecosystems," *Journal of Ethnobiology*, vol. 36, no. 2, pp. 344–360, 2016.

[38] K. Lyons, *The Decomposition of Life: Politics, Violence, and Coca*. Duke University Press, 2016.

[39] The New York Times, "The u.s. military said it helped bomb a drug camp in ecuador. it was a dairy farm.." `https://www.nytimes.com/2026/03/24/world/americas/us-ecuador-drug-camp-bombing-dairy-farm.html`, Mar. 2026. Accessed 5 April 2026.

[40] L. Floridi, J. Cowls, M. Beltrametti, *et al.*, "Ai4people—an ethical framework for a good ai society," *Minds and Machines*, vol. 28, no. 4, pp. 689–707, 2018.

[41] International Cocoa Organization, "Quarterly bulletin of cocoa statistics," tech. rep., ICCO, 2023.

[42] T. Plowman, "The ethnobotany of coca (*erythroxylum coca*)," *Journal of Ethnopharmacology*, vol. 10, no. 1, pp. 1–21, 1984.

[43] R. M. Julien and C. Advokat, *A Primer of Drug Action*. Worth Publishers, 2014.

[44] D. J. Nutt, L. A. King, and L. D. Phillips, "Drug harms in the uk: A multicriteria decision analysis," *The Lancet*, vol. 376, no. 9752, pp. 1558–1565, 2010.

[45] D. Bewley-Taylor, *International Drug Control: Consensus Fractured*. Cambridge University Press, 2012.

[46] United Nations Office on Drugs and Crime, "Alternative development: A global thematic evaluation," tech. rep., UNODC, 2015.

[47] G. Schroth and C. A. Harvey, "Biodiversity conservation in cocoa production landscapes: An overview," *Biodiversity and Conservation*, vol. 16, pp. 2237–2244, July 2007.

[48] C. Palm, H. Blanco-Canqui, F. DeClerck, L. Gatere, and P. Grace, "Conservation agriculture and ecosystem services: An overview," *Agriculture, Ecosystems & Environment*, vol. 187, pp. 87–105, Apr. 2014.

[49] P. S. Carberry, W.-l. Liang, S. Twomlow, D. P. Holzworth, J. P. Dimes, T. McClelland, N. I. Huth, F. Chen, Z. Hochman, and B. A. Keating, "Scope for improved eco-efficiency varies among diverse cropping systems," *Proceedings of the National Academy of Sciences*, vol. 110, pp. 8381–8386, May 2013.

[50] M. S. Wolfe, "Crop strength through diversity," *Nature*, vol. 406, pp. 681–682, Aug. 2000.

[51] E. Somarriba and J. Beer, "Productivity of Theobroma cacao agroforestry systems with timber or legume service shade trees," *Agroforestry Systems*, vol. 81, pp. 109–121, Feb. 2011.

[52] C. Gidoin, J. Avelino, O. Deheuvels, C. Cilas, and M. A. N. Bieng, "Shade Tree Spatial Structure and Pod Production Explain Frosty Pod Rot Intensity in Cacao Agroforests, Costa Rica," *Phytopathology®*, vol. 104, pp. 275–281, Mar. 2014.

[53] R. Cerda, J. Avelino, C. A. Harvey, C. Gary, P. Tixier, and C. Allinne, "Coffee agroforestry systems capable of reducing disease-induced yield and economic losses while providing multiple ecosystem services," *Crop Protection*, vol. 134, p. 105149, Aug. 2020.

[54] L. Pumariño, G. W. Sileshi, S. Gripenberg, R. Kaartinen, E. Barrios, M. N. Muchane, C. Midega, and M. Jonsson, "Effects of agroforestry on pest, disease and weed control: A meta-analysis," *Basic and Applied Ecology*, vol. 16, pp. 573–582, Nov. 2015.

[55] B. Utomo, A. A. Prawoto, S. Bonnet, A. Bangviwat, and S. H. Gheewala, "Environmental performance of cocoa production from monoculture and agroforestry systems in Indonesia," *Journal of Cleaner Production*, vol. 134, pp. 583–591, Oct. 2016.

[56] G. Schroth, C. A. Harvey, G. Vincent, G. A. B. da Fonseca, C. Gascon, and H. L. Vasconcelos, "Conservation in tropical landscape mosaics: The case of cacao agroforests in southeastern bahia, brazil," *Biodiversity and Conservation*, vol. 20, no. 8, pp. 1635–1654, 2011.

[57] J. H. Bowers, B. A. Bailey, P. K. Hebbar, S. Sanogo, and R. D. Lumsden, "The Impact of Plant Diseases on World Chocolate Production," *Plant Health Progress*, vol. 2, p. 12, Jan. 2001.

[58] D. Guest, "Black Pod: Diverse Pathogens with a Global Impact on Cocoa Yield," *Phytopathology®*, vol. 97, pp. 1650–1653, Dec. 2007.

[59] R. H. Fulton, "The Cacao Disease Trilogy: Black Pod, Monilia Pod Rot, and Witches'-Broom," *The American Phytopathological Society*, p. 601, 1989.

[60] M. Obiakara, P. Etaware, and K. Chukwuka, "Maximum Entropy Niche Modelling to Estimate the Potential Distribution of Phytophthora megakarya (Brasier & M. J. Griffin) in Tropical Regions," *European Journal of Ecology*, vol. 6, Dec. 2020.

[61] J. H. Nagel, M. Gryzenhout, B. Slippers, and M. J. Wingfield, "The occurrence and impact of Phytophthora on the African continent.," in *Phytophthora: A Global Perspective* (K. Lamour, ed.), pp. 204–214, Wallingford: CABI, 2013.

[62] F. Perrine-Walker, "Phytophthora palmivora–Cocoa Interaction," *Journal of Fungi*, vol. 6, p. 167, Sept. 2020.

[63] R. Ploetz, "The Impact of Diseases on Cacao Production: A Global Overview," in *Cacao Diseases: A History of Old Enemies and New Encounters* (B. A. Bailey and L. W. Meinhardt, eds.), pp. 33–59, Cham: Springer International Publishing, 2016.

[64] S. Lima, C. A. Souza, N. Patrocínio, R. Silva, R. Santos, and K. Gramacho, "Favorabilidade, distribuição e prevalência da vassoura-de-bruxa do cacaueiro no estado do espírito santo, brasil," *Agrotrópica (Itabuna)*, vol. 30, pp. 5–14, Apr. 2018.

[65] R. P. Bateman, E. Hidalgo, J. García, C. Arroyo, G. Ten Hoopen, V. Adonijah, and U. Krauss, "Application of chemical and biological agents for the management of frosty pod rot (Moniliophthora roreri) in Costa Rican cocoa (Theobroma cacao)," *Annals of Applied Biology*, vol. 147, no. 2, pp. 129–138, 2005.

[66] A. Ram, *Biology, epidemiology and control of moniliasis (Moniliophthora roreri) of cacao*. PhD thesis, Imperial Gollege of Science and Technology, Berkshire, Mar. 1989.

[67] H. C. Evans, D. F. Edwards, and M. Rodríguez, "Research on Cocoa Diseases in Ecuador: Past and Present," *PANS*, vol. 23, pp. 68–80, Mar. 1977.

[68] A. W. Leach, J. D. Mumford, and U. Krauss, "Modelling Moniliophthora roreri in Costa Rica," *Crop Protection*, vol. 21, pp. 317–326, May 2002.

[69] H. C. Evans, "Pleomorphism in Crinipellis perniciosa, causal agent of witches' broom disease of cocoa," *Transactions of the British Mycological Society*, vol. 74, pp. 515–523, June 1980.

[70] CABI, "Moniliophthora roreri (frosty pod rot)," tech. rep., CABI, Nov. 2021. CABI Compendium datasheet; accessed 9 October 2023.

[71] CABI, "Phytophthora megakarya (black pod of cocoa)," tech. rep., CABI, Nov. 2021. CABI Compendium datasheet; accessed 9 October 2023.

[72] R. Noble and E. Coventry, "Suppression of soil-borne plant diseases with composts: A review," *Biocontrol Science and Technology*, vol. 15, pp. 3–20, Feb. 2005.

[73] J. E. Hunter and R. K. Kunimoto, "Dispersal of Phytophthora palmivora Sporangia by Wind-Blown Rain," *Phytopathology*, vol. 64, pp. 202–206, 1974.

[74] M. Nkeng, B. Efombagn, B. N.L, I. Sache, and C. Cilas, "Spatio-temporal dynamics on a plot scale of cocoa black pod rot caused by Phytophthora megakarya in Cameroon," *European Journal of Plant Pathology*, vol. 147, Aug. 2016.

[75] C. A. Thorold, "Airborne Dispersal of Phytophthora palmivora, causing Black-Pod Disease of Theobroma cacao," *Nature*, vol. 170, pp. 718–719, Oct. 1952.

[76] G. M. Ten Hoopen, O. Sounigo, R. Babin, Yédé, G. Dikwe, and C. Cilas, "Spatial and temporal analysis of a Phytophthora Megakarya epidemic in a plantation in the Centre of Cameroon," tech. rep., Cocoa Producers' Alliance, 2010.

[77] H. C. Evans, "Invertebrate vectors of Phytophthora palmivora, causing black pod disease of cocoa in Ghana," *Annals of Applied Biology*, vol. 75, no. 3, pp. 331–345, 1973.

[78] A. S. Artero, J. Q. Silva, P. S. B. Albuquerque, E. A. Bressan, G. A. Leal Jr, A. M. Sebbenn, G. W. Griffith, and A. Figueira, "Spatial genetic structure and dispersal of the cacao pathogen Moniliophthora perniciosa in the Brazilian Amazon," *Plant Pathology*, vol. 66, no. 6, pp. 912–923, 2017.

[79] CABI, "Moniliophthora perniciosa (witches' broom disease of cacao)," tech. rep., CABI, Nov. 2021. CABI Compendium datasheet; accessed 9 October 2023.

[80] S. A. Rudgard and D. R. Butler, "Witches' broom disease on cocoa in Rondonia, Brazil: Pod infection in relation to pod susceptibility, wetness, inoculum, and phytosanitation," *Plant Pathology*, vol. 36, no. 4, pp. 515–522, 1987.

[81] W. Soberanis, R. Ríos, E. Arévalo, L. Zúñiga, O. Cabezas, and U. Krauss, "Increased frequency of phytosanitary pod removal in cacao (Theobroma cacao) increases yield economically in eastern Peru," *Crop Protection*, vol. 18, pp. 677–685, Dec. 1999.

[82] P. K. Hebbar, "Cacao diseases: A global perspective from an industry point of view," *Phytopathology*, vol. 97, no. 12, pp. 1658–1663, 2007.

[83] M. E. H. Matson, I. M. Small, W. E. Fry, and H. S. Judelson, "Metalaxyl Resistance in Phytophthora infestans: Assessing Role of RPA190 Gene and Diversity Within Clonal Lineages," *Phytopathology®*, vol. 105, pp. 1594–1600, Dec. 2015.

[84] E. U. Asogwa and L. N. Dongo, "Problems associated with pesticide usage and application in Nigerian cocoa production: A review," *African Journal of Agricultural Research*, vol. 4, pp. 675–683, Aug. 2009.

[85] R. Dand, *The International Cocoa Trade*. Cambridge, UK: Woodhead Publishing, 3 ed., 2010.

[86] J. Hainmueller, M. J. Hiscox, and M. Tampe, "Sustainable Development for Cocoa Farmers in Ghana," *International Growth Centre*, p. 68, Mar. 2011.

[87] International Cocoa Organization, "Progress report: Action programme on pesticides," tech. rep., International Cocoa Organization, London, UK, 2007. Presented at the 133rd Meeting of the ICCO Executive Committee, held at the EBRD Offices, London, 5–7 June 2007.

[88] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep Learning for Computer Vision: A Brief Review," *Computational Intelligence and Neuroscience*, vol. 2018, p. e7068349, Feb. 2018.

[89] B. G. Weinstein, "A computer vision for animal ecology," *Journal of Animal Ecology*, vol. 87, no. 3, pp. 533–545, 2018.

[90] S. S. Chouhan, U. P. Singh, and S. Jain, "Applications of Computer Vision in Plant Pathology: A Survey," *Archives of Computational Methods in Engineering*, vol. 27, pp. 611–632, Apr. 2020.

[91] S. Xu, J. Wang, W. Shou, T. Ngo, A.-M. Sadick, and X. Wang, "Computer Vision Techniques in Construction: A Critical Review," *Archives of Computational Methods in Engineering*, vol. 28, pp. 3383–3397, Aug. 2021.

[92] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-End Object Detection with Transformers," in *Computer Vision – ECCV 2020* (A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds.), Lecture Notes in Computer Science, (Cham), pp. 213–229, Springer International Publishing, 2020.

[93] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," June 2021.

[94] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," *CVPR*, Mar. 2022.

[95] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *Advances in Neural Information Processing Systems*, vol. 28, Curran Associates, Inc., 2015.

[96] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, 2016.

[97] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," *Advances in neural information processing systems*, vol. 29, 2016.

[98] M.-H. Guo, T.-X. Xu, J.-J. Liu, Z.-N. Liu, P.-T. Jiang, T.-J. Mu, S.-H. Zhang, R. R. Martin, M.-M. Cheng, and S.-M. Hu, "Attention mechanisms in computer vision: A survey," *Computational Visual Media*, vol. 8, pp. 331–368, Sept. 2022.

[99] J. F. Lopes, V. G. T. da Costa, D. F. Barbin, L. J. P. Cruz-Tirado, V. Baeten, and S. Barbon Junior, "Deep computer vision system for cocoa classification," *Multimedia Tools and Applications*, vol. 81, pp. 41059–41077, Nov. 2022.

[100] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated Residual Transformations for Deep Neural Networks," Apr. 2017.

[101] H. Tian, T. Wang, Y. Liu, X. Qiao, and Y. Li, "Computer vision technology in agricultural automation —A review," *Information Processing in Agriculture*, vol. 7, pp. 1–19, Mar. 2020.

[102] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, pp. 84–90, May 2017.

[103] C. Rodriguez, O. Alfaro, P. Paredes, D. Esenarro, and F. Hilario, "Machine Learning Techniques in the Detection of Cocoa (Theobroma cacao L.) Diseases," *Annals of the Romanian Society for Cell Biology*, pp. 7732–7741, Apr. 2021.

[104] D. S. Tan, R. N. Leong, A. F. Laguna, C. A. Ngo, A. Lao, D. M. Amalin, and D. G. Alvindia, "AuToDiDAC: Automated Tool for Disease Detection and Assessment for Cacao Black Pod Rot," *Crop Protection*, vol. 103, pp. 98–102, Jan. 2018.

[105] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[106] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[107] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is All you Need," in *Advances in Neural Information Processing Systems*, vol. 30, Curran Associates, Inc., 2017.

[108] D. Hendrycks and K. Gimpel, "Gaussian Error Linear Units (GELUs)," *arXiv:1606.08415*, June 2023.

[109] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," *arXiv:1502.03167*, Mar. 2015.

[110] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer Normalization," *arXiv:1607.06450*, July 2016.

[111] Y. Wu and K. He, "Group normalization," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018.

[112] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014.

[113] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection," Feb. 2018.

[114] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[115] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, pp. 886–893, 2005.

[116] T.-L. Lin, H.-Y. Chang, and K.-H. Chen, "The pest and disease identification in the growth of sweet peppers using faster r-cnn and mask r-cnn," *Journal of Internet Technology*, vol. 21, no. 2, pp. 605–614, 2020.

[117] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *Proceedings of the 32nd International Conference on Machine Learning*, pp. 448–456, PMLR, June 2015.

[118] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961–2969, 2017.

[119] H. Huang, Z. Wei, and L. Yao, "A Novel Approach to Component Assembly Inspection Based on Mask R-CNN and Support Vector Machines," *Information*, vol. 10, p. 282, Sept. 2019.

[120] S. Cao and R. Nevatia, "Exploring deep learning based solutions in fine grained activity recognition in the wild," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 384–389, Dec. 2016.

[121] A. R. Huda-Shakirah, N. M. I. Mohamed Nor, L. Zakaria, Y.-H. Leong, and M. H. Mohd, "Lasiodiplodia theobromae as a causal pathogen of leaf blight, stem canker, and pod rot of Theobroma cacao in Malaysia," *Scientific Reports*, vol. 12, p. 8966, May 2022.

[122] "Chemical control of lasiodiplodia," 2005. Placeholder entry. Please replace with full bibliographic details.

[123] M. M. Salvatore, A. Andolfi, and R. Nicoletti, "The Thin Line between Pathogenicity and Endophytism: The Case of Lasiodiplodia theobromae," *Agriculture*, vol. 10, p. 488, Oct. 2020.

[124] L. P. N. M. Kroon, H. Brouwer, A. W. A. M. de Cock, and F. Govers, "The Genus Phytophthora Anno 2012," *Phytopathology®*, vol. 102, pp. 348–364, Apr. 2012.

[125] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," in *Computer Vision – ECCV 2014* (D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, eds.), Lecture Notes in Computer Science, (Cham), pp. 740–755, Springer International Publishing, 2014.

[126] A. Horzyk and E. Ergün, "YOLOv3 Precision Improvement by the Weighted Centers of Confidence Selection," in *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, July 2020.

[127] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable Transformers for End-to-End Object Detection," *arXiv:2010.04159*, Mar. 2021.

[128] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H.-Y. Shum, "Dino: Detr with improved denoising anchor boxes for end-to-end object detection," *arXiv preprint arXiv:2203.03605*, 2022.

[129] W. Lv, Y. Zhao, Q. Chang, K. Huang, G. Wang, and Y. Liu, "RT-DETRv2: Improved baseline with bag-of-freebies for real-time detection transformer," 2024. arXiv:2407.17140.

[130] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015* (N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, eds.), Lecture Notes in Computer Science, (Cham), pp. 234–241, Springer International Publishing, 2015.

[131] A. O. Vuola, S. U. Akram, and J. Kannala, "Mask-RCNN and U-Net Ensembled for Nuclei Segmentation," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pp. 208–212, Apr. 2019.

[132] P. Bharati and A. Pramanik, "Deep Learning Techniques—R-CNN to Mask R-CNN: A Survey," in *Computational Intelligence in Pattern Recognition* (A. K. Das, J. Nayak, B. Naik, S. K. Pati, and D. Pelusi, eds.), Advances in Intelligent Systems and Computing, (Singapore), pp. 657–668, Springer, 2020.

[133] T. Zhao, Y. Yang, H. Niu, D. Wang, and Y. Chen, "Comparing U-Net convolutional network with mask R-CNN in the performances of pomegranate tree canopy segmentation," in *Multispectral, Hyperspectral, and Ultraspectral Remote Sensing Technology, Techniques and Applications VII*, vol. 10780, pp. 210–218, SPIE, Oct. 2018.

[134] C. Li and M. Wand, "Precomputed Real-Time Texture Synthesis with Markovian Generative Adversarial Networks," in *Computer Vision – ECCV 2016* (B. Leibe, J. Matas, N. Sebe, and M. Welling, eds.), Lecture Notes in Computer Science, (Cham), pp. 702–716, Springer International Publishing, 2016.

[135] D. P. Kingma and M. Welling, "An Introduction to Variational Autoencoders," *Foundations and Trends® in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019.

[136] A. Banerjee, "An Analysis of Logistic Models: Exponential Family Connections and Online Performance," in *Proceedings of the 2007 SIAM International Conference on Data Mining*, pp. 204–215, Society for Industrial and Applied Mathematics, Apr. 2007.

[137] W. Xu, H. Sun, C. Deng, and Y. Tan, "Variational Autoencoder for Semi-Supervised Text Classification," in *Thirty-First AAAI Conference on Artificial Intelligence*, Feb. 2017.

[138] W. Xu and Y. Tan, "Semisupervised Text Classification by Variational Autoencoder," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, pp. 295–308, Jan. 2020.

[139] N. Dilokthanakul, P. A. M. Mediano, M. Garnelo, M. C. H. Lee, H. Salimbeni, K. Arulkumaran, and M. Shanahan, "Deep Unsupervised Clustering with Gaussian Mixture Variational Autoencoders," Jan. 2017.

[140] K.-L. Lim, X. Jiang, and C. Yi, "Deep Clustering With Variational Autoencoder," *IEEE Signal Processing Letters*, vol. 27, pp. 231–235, 2020.

[141] J. An and S. Cho, "Variational Autoencoder based Anomaly Detection using Reconstruction Probability," *Special lecture on IE*, 2015.

[142] X. Li and J. She, "Collaborative Variational Autoencoder for Recommender Systems," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, (New York, NY, USA), pp. 305–314, Association for Computing Machinery, Aug. 2017.

[143] E. Lin, S. Mukherjee, and S. Kannan, "A deep adversarial variational autoencoder model for dimensionality reduction in single-cell RNA sequencing analysis," *BMC Bioinformatics*, vol. 21, p. 64, Feb. 2020.

[144] Y. Fan, G. Wen, D. Li, S. Qiu, M. D. Levine, and F. Xiao, "Video anomaly detection and localization via Gaussian Mixture Fully Convolutional Variational Autoencoder," *Computer Vision and Image Understanding*, vol. 195, p. 102920, June 2020.

[145] A. Vahdat and J. Kautz, "NVAE: A Deep Hierarchical Variational Autoencoder," Jan. 2021.

[146] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep Learning Face Attributes in the Wild," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3730–3738, 2015.

[147] A. Asperti, D. Evangelista, and E. Loli Piccolomini, "A Survey on Variational Autoencoders from a Green AI Perspective," *SN Computer Science*, vol. 2, p. 301, May 2021.

[148] J. Sun, X. Wang, N. Xiong, and J. Shao, "Learning Sparse Representation With Variational Auto-Encoder for Anomaly Detection," *IEEE Access*, vol. 6, pp. 33353–33361, 2018.

[149] L. Maaløe, M. Fraccaro, V. Liévin, and O. Winther, "BIVA: A Very Deep Hierarchy of Latent Variables for Generative Modeling," in *Advances in Neural Information Processing Systems*, vol. 32, Curran Associates, Inc., 2019.

[150] E. Nalisnick, A. Matsukawa, Y. W. Teh, D. Gorur, and B. Lakshminarayanan, "Do Deep Generative Models Know What They Don't Know?," Feb. 2019.

[151] X. Zhu, *Semi-Supervised Learning Literature Survey.* PhD thesis, University of Wisconsin, Madison, 2005.

[152] J. E. van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine Learning*, vol. 109, pp. 373–440, Feb. 2020.

[153] M. Valdenegro-Toro and D. S. Mori, "A Deeper Look into Aleatoric and Epistemic Uncertainty Disentanglement," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1508–1516, June 2022.

[154] T. Pearce, A. Brintrup, and J. Zhu, "Understanding Softmax Confidence and Uncertainty," June 2021.

[155] M. Amer and T. Maul, "A Review of Modularization Techniques in Artificial Neural Networks," *Artificial Intelligence Review*, vol. 52, pp. 527–561, June 2019.

[156] "Modularity concept," 2001. Placeholder entry. Please replace with full bibliographic details.

[157] O. M. Shir, "Niching in Evolutionary Algorithms," in *Handbook of Natural Computing* (G. Rozenberg, T. Bäck, and J. N. Kok, eds.), pp. 1035–1069, Berlin, Heidelberg: Springer, 2012.

[158] D.-C. Dang, A. Eremeev, and P. K. Lehre, "Escaping Local Optima with Non-Elitist Evolutionary Algorithms," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 12275–12283, May 2021.

[159] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A Novel Image Dataset for Benchmarking Machine Learning Algorithms," Sept. 2017.

[160] Y. Sun, B. Xue, M. Zhang, and G. G. Yen, "Evolving Deep Convolutional Neural Networks for Image Classification," *IEEE Transactions on Evolutionary Computation*, vol. 24, pp. 394–407, Apr. 2020.

[161] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems*, vol. 32, Curran Associates, Inc., 2019.

[162] PyTorch Contributors, "Torchvision model documentation: VGG16." `https://pytorch.org/vision/stable/models/generated/torchvision.models.vgg16.html`, 2026. Accessed 6 April 2026.

[163] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," Sept. 2020.

[164] Y. Gong, Y. Sun, D. Peng, P. Chen, Z. Yan, and K. Yang, "Analyze COVID-19 CT images based on evolutionary algorithm with dynamic searching space," *Complex & Intelligent Systems*, vol. 7, pp. 3195–3209, Dec. 2021.

[165] D. Singh, N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, "PlantDoc: A Dataset for Visual Plant Disease Detection," in *Proceedings of the 7th ACM IKDD CoDS and*

25th COMAD, CoDS COMAD 2020, (New York, NY, USA), pp. 249–253, Association for Computing Machinery, Jan. 2020.

[166] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, Feb. 2018.

[167] M. Motamedi, N. Sakharnykh, and T. Kaldewey, "A Data-Centric Approach for Training Deep Neural Networks with Less Data," Oct. 2021.

[168] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, June 2009.

[169] J. S. Serrano Arenas and C. A. Torres Villamizar, "Cocoa diseases (yolov4): Monilia & phytophthora (diseases in cocoa pods)." `https://www.kaggle.com/datasets/serranosebas/enfermedades-cacao-yolov4`, 2020. Accessed: 2025-05-13.

[170] R. D. King, O. I. Orhobor, and C. C. Taylor, "Cross-validation is safe to use," *Nature Machine Intelligence*, vol. 3, pp. 276–276, Apr. 2021.

[171] D. Bickford, D. J. Lohman, N. S. Sodhi, P. K. L. Ng, R. Meier, K. Winker, K. K. Ingram, and I. Das, "Cryptic species as a window on diversity and conservation," *Trends in Ecology & Evolution*, vol. 22, pp. 148–155, Mar. 2007.

[172] O. Ovaskainen, J. Nokso-Koivisto, J. Hottola, T. Rajala, T. Pennanen, H. Ali-Kovero, O. Miettinen, P. Oinonen, P. Auvinen, L. Paulin, K.-H. Larsson, and R. Mäkipää, "Identifying wood-inhabiting fungi with 454 sequencing – what is the probability that BLAST gives the correct species?," *Fungal Ecology*, vol. 3, pp. 274–283, Nov. 2010.

[173] K. O'Donnell, T. J. Ward, V. A. R. G. Robert, P. W. Crous, D. M. Geiser, and S. Kang, "DNA sequence-based identification of Fusarium: Current status and future directions," *Phytoparasitica*, vol. 43, pp. 583–595, Nov. 2015.

[174] G. M. Boratyn, C. Camacho, P. S. Cooper, G. Coulouris, A. Fong, N. Ma, T. L. Madden, W. T. Matten, S. D. McGinnis, Y. Merezhuk, Y. Raytselis, E. W. Sayers, T. Tao, J. Ye, and I. Zaretskaya, "BLAST: A more efficient report with usability improvements," *Nucleic Acids Research*, vol. 41, pp. W29–W33, July 2013.

[175] T. R. Horton and T. D. Bruns, "The molecular revolution in ectomycorrhizal ecology: Peeking into the black-box," *Molecular Ecology*, vol. 10, no. 8, pp. 1855–1871, 2001.

[176] N. Luchi, R. Ioos, and A. Santini, "Fast and reliable molecular methods to detect fungal pathogens in woody plants," *Applied Microbiology and Biotechnology*, vol. 104, pp. 2453–2468, Mar. 2020.

[177] M. Ray, A. Ray, S. Dash, A. Mishra, K. G. Achary, S. Nayak, and S. Singh, "Fungal disease detection in plants: Traditional assays, novel diagnostic techniques and biosensors," *Biosensors and Bioelectronics*, vol. 87, pp. 708–723, Jan. 2017.

[178] N. W. Schaad and R. D. Frederick, "Real-time PCR and its application for rapid plant disease diagnostics," *Canadian Journal of Plant Pathology*, vol. 24, pp. 250–258, Sept. 2002.

[179] P. Horevaj, E. A. Milus, and B. H. Bluhm, "A real-time qPCR assay to quantify Fusarium graminearum biomass in wheat kernels," *Journal of Applied Microbiology*, vol. 111, no. 2, pp. 396–406, 2011.

[180] S. Goodwin, J. Gurtowski, S. Ethe-Sayers, P. Deshpande, M. C. Schatz, and W. R. McCombie, "Oxford Nanopore sequencing, hybrid error correction, and de novo assembly of a eukaryotic genome," *Genome Research*, vol. 25, pp. 1750–1756, Nov. 2015.

[181] S. Kuhlgert, G. Austic, R. Zegarac, I. Osei-Bonsu, D. Hoh, M. I. Chilvers, M. G. Roth, K. Bi, D. TerAvest, P. Weebadde, and D. M. Kramer, "MultispeQ Beta: A tool for large-scale plant phenotyping connected to the open PhotosynQ network," *Royal Society Open Science*, vol. 3, no. 10, p. 160592, 2022.

[182] C. H. Bock, G. H. Poole, P. E. Parker, and T. R. Gottwald, "Plant Disease Severity Estimated Visually, by Digital Photography and Image Analysis, and by Hyperspectral Imaging," *Critical Reviews in Plant Sciences*, vol. 29, pp. 59–107, Mar. 2010.

[183] A. F. H. Goetz, G. Vane, J. E. Solomon, and B. N. Rock, "Imaging Spectrometry for Earth Remote Sensing," *Science*, vol. 228, pp. 1147–1153, June 1985.

[184] G. Lu and B. Fei, "Medical hyperspectral imaging: A review," *Journal of Biomedical Optics*, vol. 19, p. 010901, Jan. 2014.

[185] S. Gutiérrez, A. Wendel, and J. Underwood, "Ground based hyperspectral imaging for extensive mango yield estimation," *Computers and Electronics in Agriculture*, vol. 157, pp. 126–135, Feb. 2019.

[186] B. Li, X. Xu, L. Zhang, J. Han, C. Bian, G. Li, J. Liu, and L. Jin, "Above-ground biomass estimation and yield prediction in potato by using UAV-based RGB and hyperspectral imaging," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 162, pp. 161–172, Apr. 2020.

[187] L. Feng, S. Zhu, C. Zhang, Y. Bao, X. Feng, and Y. He, "Identification of Maize Kernel Vigor under Different Accelerated Aging Times Using Hyperspectral Imaging," *Molecules*, vol. 23, p. 3078, Dec. 2018.

[188] H. Okamoto, T. Murata, T. Kataoka, and S.-I. Hata, "Plant classification for weed detection using hyperspectral imaging with wavelet analysis," *Weed Biology and Management*, vol. 7, no. 1, pp. 31–37, 2007.

[189] H. D. D. Nguyen, V. Pan, C. Pham, R. Valdez, K. Doan, and C. Nansen, "Night-based hyperspectral imaging to study association of horticultural crop leaf reflectance and nutrient status," *Computers and Electronics in Agriculture*, vol. 173, p. 105458, June 2020.

[190] T.-t. Pan, E. Chyngyz, D.-W. Sun, J. Paliwal, and H. Pu, "Pathogenetic process monitoring and early detection of pear black spot disease caused by Alternaria alternata using hyperspectral imaging," *Postharvest Biology and Technology*, vol. 154, pp. 96–104, Aug. 2019.

[191] J. Yue, W. Zhao, S. Mao, and H. Liu, "Spectral–spatial classification of hyperspectral images using deep convolutional neural networks," *Remote Sensing Letters*, vol. 6, pp. 468–477, June 2015.

[192] M. E. Paoletti, J. M. Haut, J. Plaza, and A. Plaza, "Deep learning classifiers for hyperspectral imaging: A review," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 158, pp. 279–317, Dec. 2019.

[193] B. Schölkopf, F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio, "Toward Causal Representation Learning," *Proceedings of the IEEE*, vol. 109, pp. 612–634, May 2021.

[194] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," in *Advances in Neural Information Processing Systems*, vol. 27, Curran Associates, Inc., 2014.

[195] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," in *Computer Vision – ECCV 2014* (D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, eds.), Lecture Notes in Computer Science, (Cham), pp. 818–833, Springer International Publishing, 2014.

[196] Y. Toda and F. Okura, "How Convolutional Neural Networks Diagnose Plant Disease," *Plant Phenomics*, vol. 2019, Mar. 2019.

[197] B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto, and F. Roli, "Evasion Attacks against Machine Learning at Test Time," in *Advanced Information Systems Engineering* (D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, C. Salinesi, M. C. Norrie, and Ó. Pastor, eds.), vol. 7908, pp. 387–402, Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.

[198] J. Peters, D. Janzing, and B. Schölkopf, *Elements of Causal Inference: Foundations and Learning Algorithms*. The MIT Press, 2017.

[199] N. O'Mahony, S. Campbell, A. Carvalho, S. Harapanahalli, G. V. Hernandez, L. Krpalkova, D. Riordan, and J. Walsh, "Deep Learning vs. Traditional Computer Vision," in *Advances in Computer Vision* (K. Arai and S. Kapoor, eds.), Advances in Intelligent Systems and Computing, (Cham), pp. 128–144, Springer International Publishing, 2020.

[200] E. Chelebian, C. Avenel, K. Kartasalo, M. Marklund, A. Tanoglidi, T. Mirtti, R. Colling, A. Erickson, A. D. Lamb, J. Lundeberg, and C. Wählby, "Morphological Features Extracted by AI Associated with Spatial Transcriptomics in Prostate Cancer," *Cancers*, vol. 13, p. 4837, Jan. 2021.

[201] X. Yang and R. P. McCord, "CoSTA: Unsupervised Convolutional Neural Network Learning for Spatial Transcriptomics Analysis," *bioRxiv*, p. 26, 2021.

[202] K.-H. Yu, F. Wang, G. J. Berry, C. Ré, R. B. Altman, M. Snyder, and I. S. Kohane, "Classifying non-small cell lung cancer types and transcriptomic subtypes using convolutional neural networks," *Journal of the American Medical Informatics Association*, vol. 27, pp. 757–769, May 2020.

[203] J. G. Meyer, S. Liu, I. J. Miller, J. J. Coon, and A. Gitter, "Learning Drug Functions from Chemical Structures with Convolutional Neural Networks and Random Forests," *Journal of Chemical Information and Modeling*, vol. 59, pp. 4438–4449, Oct. 2019.

[204] D. MacLean, "A convolutional neural network for predicting transcriptional regulators of genes in Arabidopsis transcriptome data reveals classification based on positive regulatory interactions," Apr. 2019.

[205] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations From Deep Networks via Gradient-Based Localization," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 618–626, 2017.

[206] L. Wang, Z. Wu, S. Karanam, K.-C. Peng, and R. V. Singh, "Reducing visual confusion with discriminative attention," *arXiv preprint arXiv:1811.07484*, 2018. Accessed: 2025-05-23.

[207] S. Ghosal, D. Blystone, A. K. Singh, B. Ganapathysubramanian, A. Singh, and S. Sarkar, "An explainable deep machine vision framework for plant stress phenotyping," *Proceedings of the National Academy of Sciences*, vol. 115, pp. 4613–4618, May 2018.

[208] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, "Searching for MobileNetV3," Nov. 2019.

[209] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga,

S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," Mar. 2016.

[210] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. VanderPlas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[211] K. Solodskikh, A. Kurbanov, R. Aydarkhanov, I. Zhelavskaya, Y. Parfenov, D. Song, and S. Lefkimmiatis, "Integral Neural Networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16113–16122, 2023.

[212] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2." `https://github.com/facebookresearch/detectron2`, 2019. Accessed: 2023-10-28.

[213] G. Jocher and the Ultralytics team, "Ultralytics models documentation." `https://docs.ultralytics.com/models/`, 2024. Accessed: 2025-05-30.

[214] G. Geetharamani and A. Pandian, "Identification of plant leaf diseases using a nine-layer deep convolutional neural network," *Computers & Electrical Engineering*, vol. 76, pp. 323–338, June 2019.

[215] R. Thapa, K. Zhang, N. Snavely, S. Belongie, and A. Khan, "The Plant Pathology Challenge 2020 data set to classify foliar disease of apples," *Applications in Plant Sciences*, vol. 8, no. 9, p. e11390, 2020.

[216] United Nations Department of Economic and Social Affairs, Population Division, "World population prospects 2022: Summary of results." `https://population.un.org/wpp/`, 2022. Accessed: 2022-05-18.

[217] L. Biewald, "Experiment Tracking with Weights and Biases." https://www.wandb.com/, 2020.

[218] J. G. A. Barbedo, L. V. Koenigkan, and T. T. Santos, "Identifying multiple plant diseases using digital image processing," *Biosystems Engineering*, vol. 147, pp. 104–116, July 2016.

[219] J. G. A. Barbedo, "A review on the main challenges in automatic plant disease identification based on visible range images," *Biosystems Engineering*, vol. 144, pp. 52–60, Apr. 2016.

[220] K. Steddom, M. McMullen, B. Schatz, and C. M. Rush, "Comparing Image Format and Resolution for Assessment of Foliar Diseases of Wheat," *Plant Health Progress*, vol. 6, p. 11, Jan. 2005.

[221] U. Krauss, "Moniliophthora roreri (frosty pod rot)." `https://www.cabidigitallibrary.org/doi/full/10.1079/cabicompendium.34779`, 2012. Accessed: 2023-10-09.

[222] CABI, "Phytophthora megakarya (black pod of cocoa)." CABI Compendium, Nov. 2021. Available at: `https://www.cabidigitallibrary.org/doi/10.1079/cabicompendium.40979` (Accessed: 9 October 2023).

[223] J. R. Sykes, K. J. Denby, and D. W. Franks, "Computer vision for plant pathology: A review with examples from cocoa agriculture," *Applications in Plant Sciences*, vol. 12, no. 2, p. e11559, 2024.

[224] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-Excitation Networks," *arXiv:1709.01507*, May 2019.

[225] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Weinberger, "Deep Networks with Stochastic Depth," *arXiv:1603.09382*, July 2016.

[226] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv:1207.0580*, July 2012.

[227] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021.

[228] S. Woo, S. Debnath, R. Hu, X. Chen, Z. Liu, I. S. Kweon, and S. Xie, "ConvNeXt V2: Co-Designing and Scaling ConvNets With Masked Autoencoders," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16133–16142, 2023.

[229] K. Wu, J. Zhang, H. Peng, M. Liu, B. Xiao, J. Fu, and L. Yuan, "TinyViT: Fast Pretraining Distillation for Small Vision Transformers," July 2022.

[230] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial Feature Learning," *arXiv:1605.09782*, Apr. 2017.

[231] T. DeVries and G. W. Taylor, "Improved Regularization of Convolutional Neural Networks with Cutout," *arXiv:1708.04552*, Nov. 2017.

[232] S. Goyal, A. Raghunathan, M. Jain, H. V. Simhadri, and P. Jain, "DROCC: Deep Robust One-Class Classification," in *Proceedings of the 37th International Conference on Machine Learning*, pp. 3711–3721, PMLR, Nov. 2020.

[233] C. Walshaw and M. G. Everett, "Multilevel landscapes in combinatorial optimisation," Tech. Rep. 02/IM/93, University of Greenwich, Computing and Mathematical Sciences, London, UK, 2002.

[234] D. B. Kell, "Scientific discovery as a combinatorial optimisation problem: How best to navigate the landscape of possible experiments?," *BioEssays*, vol. 34, no. 3, pp. 236–244, 2012.

[235] M. C. Aime and W. Phillips-Mora, "The causal agents of witches' broom and frosty pod rot of cacao (chocolate, theobroma cacao) form a new lineage of marasmiaceae," *Mycologia*, vol. 97, no. 5, pp. 1012–1022, 2005.

[236] H. C. Evans, J. A. Stalpers, R. A. Samson, and G. L. Benny, "On the taxonomy of monilia roreri, an important pathogen of theobroma cacao in south america," *Canadian Journal of Botany*, vol. 56, no. 20, pp. 2528–2532, 1978.

[237] A. Y. Akrofi, "Phytophthora megakarya: A review on its status as a pathogen on cacao in west africa," *African Crop Science Journal*, vol. 23, pp. 67–87, Mar. 2015.

[238] B. A. Bailey, H. C. Evans, W. Phillips-Mora, S. S. Ali, and L. W. Meinhardt, "Moniliophthora roreri, causal agent of cacao frosty pod rot," *Molecular Plant Pathology*, vol. 19, no. 7, pp. 1580–1594, 2018.

[239] A. Shamsian, O. Kleinfeld, A. Globerson, and G. Chechik, "Learning Object Permanence from Video," in *Computer Vision – ECCV 2020* (A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds.), (Cham), pp. 35–50, Springer International Publishing, 2020.

[240] J. R. Sykes, K. J. Denby, and D. W. Franks, "Tailoring convolutional neural networks for custom botanical data," *Applications in Plant Sciences*, p. e11620, 2025.

[241] M. Guillaumin, J. Verbeek, and C. Schmid, "Multimodal semi-supervised learning for image classification," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 902–909, June 2010.

[242] Z. Liu, M. Sun, T. Zhou, G. Huang, and T. Darrell, "Rethinking the Value of Network Pruning," Mar. 2019.

[243] J.-Y. Wu, C. Yu, S.-W. Fu, C.-T. Liu, S.-Y. Chien, and Y. Tsao, "Increasing Compactness of Deep Learning Based Speech Enhancement Models With Parameter Pruning and Quantization Techniques," *IEEE Signal Processing Letters*, vol. 26, pp. 1887–1891, Dec. 2019.