Article:
Sykes, Jamie, Denby, Katherine orcid.org/0000-0002-7857-6814 and Franks, Daniel Wayne orcid.org/0000-0002-4832-7470 (2024) Computer vision for plant pathology:A review with examples from cocoa agriculture. Applications in Plant Sciences. e11559. ISSN: 2168-0450

https://doi.org/10.1002/aps3.11559

Computer vision for plant pathology: A review with examples from cocoa agriculture

Jamie R. Sykes[a,1], Katherine J. Denby[b], Daniel W. Franks[a,c]

[a] *Department of Computer Science, University of York, Deramore Lane, York, YO10 5GH, Yorkshire, UK*
[b] *Centre for Novel Agricultural Products, Department of Biology, University of York, Wentworth Way, York, YO10 5DD, Yorkshire, UK*
[c] *Department of Biology, University of York, Wentworth Way, York, YO10 5DD, Yorkshire, UK*

ABSTRACT

Plant pathogens can decimate crops and render the local cultivation of a species unprofitable. In extreme cases this has caused famine and economic collapse. Timing is vital in treating crop diseases and use of computer vision for precise disease detection and timing of pesticide application is gaining popularity. Computer vision can reduce labour costs, prevent misdiagnosis of disease and prevent misapplication of pesticides. Pesticide misapplication is both financially costly and can exacerbate pesticide resistance and pollution. Here we review the application and development of computer vision and machine learning methods for detection of plant disease. This review goes beyond the scope of previous works to discuss important technical concepts and considerations when applying computer vision to plant pathology. We present new case studies on adapting standard computer vision methods and we review techniques for training data acquisition, use of diag-

Email addresses: jamie.sykes@york.ac.uk (Jamie R. Sykes), katherine.denby@york.ac.uk (Katherine J. Denby), daniel.franks@york.ac.uk (Daniel W. Franks)

nostic tools from biology and inspection of informative features. In addition to in-depth discussion of convolutional neural networks and transformers, we also highlight the strengths of methods such as support vector machines and evolved neural networks. We discuss the benefits of carefully curating training data and situations where less computationally expensive techniques are advantageous. This includes a comparison of popular model architectures and a guide to their implementation.

Keywords: agronomy; disease detection; machine learning; plant pathology

Manuscript received _____; revision accepted _____.

INTRODUCTION

Computer vision (CV), typically powered by machine learning (ML), is now used for a variety of tasks in agriculture, botany and ecology. These tasks include plant health assessments (Patrício and Rieder, 2018), identification of weeds (Wu et al., 2021), identification of drought prone areas of land (Ramos-Giraldo et al., 2020), yield prediction (Sarkate et al., 2013) and detection of defects or bruising in fruits and vegetables (Tripathi and Maktedar, 2020). We are seeing substantial improvement in the efficiency of CV techniques (He et al., 2016; Howard et al., 2017; Zhang et al., 2018) and, at least for now, computational resources continue to become cheaper (Mack, 2011). As a result, CV is becoming available to whole industries, not just areas of highest commercial value. For example, ML has been used with increasing regularity for cocoa specific tasks such as the exploration and optimisation of aroma profiles (Fuentes et al., 2019), monitoring of cocoa bean fermentation (Parra et al., 2018; Oliveira et al., 2021) and bean quality classification

(Mite-Baidal et al., 2019). Large research and development budgets for areas like wheat production have allowed for the use of unpiloted aerial vehicle photography to identify disease outbreaks (Su et al., 2018; Chiu et al., 2020) and the use of multispectral satellite photography to monitor outbreaks of yellow rust from space (Nagarajan et al., 1984). Yet the application of ML to sectors with fewer financial resources has had to take a different form. Onboard GPUs can run large neural networks, analysing image data from farm machinery in real time locally and fast internet connections can be used to run the same large models remotely (Grosch, 2018). However, implementation in poorer sectors must rely on older hardware, edge devices and older model smartphones. This means that an emphasis must be placed on ultra low cost implementation and high computational efficiency of algorithms. This provides us with an opportunity and motivation to steer the ML field away from brute force computing and towards more nuanced and efficient approaches.

The cultivation of cocoa, *Theobroma cacao*, represents a prime example of a sector that could benefit greatly from non-intrusive and highly optimised CV disease detection and will be used as an example throughout this review. 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 (Maddison et al., 1995; Marelli et al., 2019). Additionally, international disease spread has been devastating to this industry in the past and could be again in the future (Phillips-Mora and Wilkinson, 2007; Meinhardt et al., 2008). Following the loss of a cocoa crop to witches' broom disease, a plot of land will typically be

cleared of forest and the previous robust agroforestry system will be replaced with a monoculture (Rice and Greenberg, 2000; Meinhardt et al., 2008). This disease is therefore not only capable of devastating the livelihoods of whole communities of cocoa farmers, eliminating 50-90% of their crop (Meinhardt et al., 2008), but it is also destructive to local biodiversity and has significant negative impact on the carbon capture potential of the land (Kuok Ho and Yap, 2020). Such loss of amazonian forest is a driver of climate change, causing positive feedback, exacerbating this global crisis (Malhi et al., 2008).

A review from 1986 on the use of systemic fungicides to tackle oomycetes, like *Phytophthora spp.*, highlights the concern about damage to the environment and human health by pesticides such as methyl bromide, which are still in use (Cohen and Coffey, 1986). These concerns and those of the pesticide resistance (Department of Health. Victoria, 2023) are still present 37 years later. However, the use of CV and ML for targeted application and calibration of pesticide dose are beginning to have massive beneficial effects in this area across the agriculture industry. It is estimated that from 2016 to 2026 smartphone use will have gone from approximately 3.7 billion people to 7.5 billion (Statista, 2022). Therefore, the necessary hardware to run CV models is largely in place and we need now only develop and deploy the CV models to have great potential for impact with little monetary input. Here we discuss how best to do that.

This review is composed of three main sections. Section one critically reviews a wide variety of relevant techniques in ML and CV model development and testing, and section two discusses techniques for data gathering, data labeling and

model testing. While section one focuses on ML theory and comparison of model architectures, section two focuses on more practical issues. Finally, section three discuses a brief roadmap to commercial implementation, which includes multiple points that are important to consider prior to choosing an architecture and beginning development.

There are several review articles published on the topic of computer vision and deep learning that are applicable to plant pathology (Voulodimos et al., 2018; Weinstein, 2018; Chouhan et al., 2020; Xu et al., 2021). High quality works such as Weinstein (2018), which reviews the use of CV in animal ecology, are directly applicable to plant pathology owing to the flexibility of the techniques discussed here. What is missing from these works is a critical review and discussion of the latest and/or less conventional techniques in CV and discussion of data acquisition and validation. Each of the aforementioned reviews were published prior or near to the release of DETR (Carion et al., 2020), VIT (Dosovitskiy et al., 2021) and ConvNeXT (Liu et al., 2022). So naturally these recent landmark methods are not discussed. However despite all being published after the release of Faster-RCNN (Ren et al., 2015), ResNet (He et al., 2016) and YOLO (Redmon et al., 2016), only Xu et al. (2021) mentions any of these popular and high performing architectures. Those being YOLO and region-based fully convolutional networks, an early predecessor to Faster-RCNN.

A recent survey (Guo et al., 2022), 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 SVMs (Lopes et al., 2022), while another recent review gives a high level discussion of a number of CV studies in agriculture, covering topics of hyper-spectra imaging, use of unpiloted areal vehicles and architectures as recent as ResNeXt (Xie et al., 2017; Tian et al., 2020). 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 Lopes et al. (2022) or Tian et al. (2020) 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 make the present review unique among those listed above.

This review provides the reader with an in-depth understanding of computer vision for plant pathology and supports the previous aforementioned 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 related fields such as plant and animal ecology and forestry.

1. METHODS IN COMPUTER VISION

1.1. Background

Ever since AlexNet was presented at NeurIPS in 2012, the field of computer vision has been dominated by convolutional neural networks (CNNs) (Krizhevsky et al., 2017). While subsequent updates to CNN architectures have provided dramatic improvements over AlexNet (Liu et al., 2022), 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 support vector machines (SVM), random forest regression and artificial neural networks to identify common diseases in cocoa from standard colour images, hereafter referred to as RGB (Red, Green, Blue) images (Rodriguez et al., 2021). 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 a SVM to perform pixel-wise identification of black pod rot in cocoa (Tan et al., 2018). 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 is 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.

1.2. Vision transformers

In the early 2010's transformers become the default for natural language processing (Liu et al., 2022) and they are now rapidly gaining popularity in vision based

tasks. Pure transformer based multilayer perceptrons, such as Vision Transformer (Dosovitskiy et al., 2021), do away with the convolutional layers of a CNN. Instead they subdivide and tokenise an image before passing this data to the fully connected layers of a network. The main drawbacks of such transformer based models are that they require training datasets on the order of millions of images and they lack the inductive biases of CNNs, like translational equivariance (Dosovitskiy et al., 2021). 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* dataset of tens of thousands of images, vision transformers can out-compete CNNs (Dosovitskiy et al., 2021).

Although the requirement for vast training datasets 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 have been 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 (Liu et al., 2022). 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 section 1.7. However, as the ConvNeXt architectures are relatively large in size (ConvNeXt-tiny: 29 million parameters, ResNet18: 12 million parame-

ters, ResNet50: 26 million parameters), these models too require large and/or complex training data sets to avoid overfitting and more powerful hardware to run at inference than the smaller ResNets.

1.3. 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. For an example of semantic segmentation, see Case Study One below.

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. 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 R-CNN to detect human visible signs of insect damage in sweet peppers, Faster R-CNN was shown to have superior accuracy and mean average precision (mAP) (Lin et al., 2020). 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) 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 (Lin et al., 2020). Though 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 Lin et al. (2020) should be considered accordingly.

Object detection and semantic segmentation are typically performed using either Faster R-CNN (Ren et al., 2015), Mask R-CNN (He et al., 2017) or YOLO (Redmon et al., 2016). 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 (Voulodimos et al., 2018). For example, SVMs have been used in conjunction with Mask R-CNN in automated ML pipelines to identify defects in machined parts (Huang et al., 2019). 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 (Cao and Nevatia, 2016). This may prove useful in projects with few training images or when classifying images of plant disease with similar characteristics such as black pod rot 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 know to infect *T. cacao*, can present with identical morphological characteristics. This means that traditional classification techniques are insufficient and molecular identification techniques must be used in their place (Huda-Shakirah et al., 2022). CV technologies that can make such difficult distinctions would have important implications for all areas of agriculture and botany for two reasons; 1) While *Phytophthora megakarya* and *Phytophthora palmivora* are managed in the same way, different species of *Lasiodiplodia* are not (Khanzada et al., 2005). Thus, 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. 2) 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 (Salvatore et al., 2020), while the growing list of *Phytophthora* (aka "plant destroyer") species described is currently 116 entries long (Kroon et al., 2012).

Transformer-based object detection models such as Detection Transformer (DETR) (Carion et al., 2020) are also now available and contend well with Faster R-CNN when trained on the huge COCO dataset. 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 (Horzyk and Ergün, 2020).

However, DETR has problems detecting small objects, and has a very long convergence time. These defects are said to be resolved in Deformable DETR (Zhu et al., 2021), though we encountered significant difficulty in retraining Deformable-DETR due to prevailing bugs in the code and so were unable to confirm these benefits. In segmenting instances of nuclei in microscopy images, Mask R-CNN was compared with the U-Net architecture, which was designed for medical image segmentation. Here the two techniques were shown to give similar mAP, F1 and recall scores (Vuola et al., 2019). However, 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) compared to 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 could have the highest F1, yet 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 (Bharati and Pramanik, 2020; Horzyk and Ergün, 2020). Additionally, in a study comparing the use of U-Net and Mask R-CNN to segment images of pomegranate trees, Mask R-CNN outperformed the U-Net in both precision and recall by wide margins (Zhao et al., 2018).

An alternative approach applied an SVM to perform pixel-wise classification to detect black pod rot in cocoa and used a human expert to label diseased pixels in training images (Tan et al., 2018). Like semantic segmentation, this technique achieves the effect of providing the model with additional information on the location of disease in an image, relative to a simple CNN. However it imposes arbitrary physical boundaries around disease symptoms such as lesions and cankers and the algorithm is unable to define for itself any symptoms that aren't or can't 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. *i.e.* the model would be able to detect non-human visible symptoms via feature learning and model the effects of hyphae propagating through the plant or systemic changes to a plant's phenotype away from the site of infection.

Case study one: Semantic segmentation for cocoa disease detection

In this case study we applied 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 case study were somewhat encouraging. However, although the selected positive results in figure one 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 above, manual annotation of a full training dataset with masks is extremely laborious. So without the promise of improved results, relative to a simple CNN, this additional effort may not pay. However, considering the favourable preliminary results in this study and one other (Zhao et al., 2018), with the incorporation of automated annotation tools and/or semi-supervised learning, semantic segmentation shows promise as an avenue of research for CV in plant pathology.

1.4. Variational autoencoders for outlier detection

In addition to discriminative modelling, ML provides several powerful tools for generative modelling. Modelling with generative deep neural networks (DNNs) 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 (Li and Wand, 2016), 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 dataset filtering.

When working with autonomously collected data, for example from camera traps or webscraping 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 lot of uninformative data and the data from naive webscraping bots can be badly contaminated with miss-classified and irrelevant images. For example, a search for the keyword "Acer" will return many more images of laptops than it will Japaneses maple trees and a search for "black pod rot" will include many images frosty pod rot, 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 labeling cannot be overstated. However, manual labeling of a full dataset 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 which are trained in parallel. 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 (Kingma and Welling, 2019). However, deep generative models are typically considered excessive for classification problems, they often have higher bias (Banerjee, 2007) and are computationally expensive.

VAEs have been used successfully for text classification (Xu et al., 2017; Xu and

Tan, 2020), data clustering (Dilokthanakul et al., 2017; Lim et al., 2020), anomaly detection (An and Cho, 2015), recommender systems (Li and She, 2017), dimensionality reduction (Lin et al., 2020) and there are published papers on the use of VAEs for anomaly detection with colour images (Fan et al., 2020), though not many.

Here we consider two methods by which a VAE might be used to detect outlying data in collections of large colour images. To do so we will use the example of detecting non-plant images in a webscraped collection of plant images for use in building a disease classifier.

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 dataset 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 dataset, compress the images in the contaminated dataset and record the values of the latent space for each image. Reduce the dimensions of the latent space further with principal component analysis, t-SNE and/or UMAP. Plot this reduced data. Outliers/contaminant images may then separate from the clean data.

Nouveau VAE (NVAE) is the product of an effort to carefully craft the encoding network architecture of a VAE and appears to produce excellent results (Vahdat and Kautz, 2021). After training for just one epoch, this architecture is able to 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 (Liu et al., 2015), 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 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 KL divergence (Asperti et al., 2021). 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 is able to reconstruct an image so well after just one training epoch, with no pre-trained weights, as the dimensionality of the data is expanded, not compressed. Likewise, NVAE is not appropriate for identifying outliers by the distribution of reconstruction errors as it can reconstruct any image almost perfectly. For example, when we trained NVAE on a dataset of 54,124 plant images, it was able to reconstruct any image in the ImageNet dataset with similar binary cross-entropy loss to that of plant images.

As an alternative to NVAE, we attempted to use a custom convolutional VAE with a ResNet-152 (He et al., 2016) backbone to apply the two methods of outlier detection described above. However, we were unable to get this architecture to function well enough to sufficiently compress the data and reconstruct images with high fidelity.

Case study two: Semi-supervised learning for outlier detection

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 a manually curated dataset of 57,228 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 the contaminated dataset of 96,692 images.

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 none-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 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.

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 (Sun et al., 2018) has largely been resolved, though for many GPU memory availability will still preclude this technique. Thus far the inability of the VAE architecture to learn a compression algorithm for large colour images suggests a hard physical limitation that might not be overcome. Moreover, while Maalø et al. (2019) contest this argument, Nalisnick et al. (2019) argue comprehensively that generative models are not suitable for outlier

393 detection by the reconstruction loss method described above as these models tend
394 to learn low-level statistics about data rather than high-level semantics. As such
395 they are often unable to differentiate between images that, to the human eye, are
396 obviously different.

397 1.5. Evolutionary algorithms

398 The field of CV is currently dominated by handcrafted DNNs with fixed topolo-
399 gies. However, the seldom used techniques of evolved neural networks have real
400 potential in the field of plant pathology. Computational efficiency at inference and
401 improved ability to generalise is of paramount importance to models developed
402 for plant pathology in the field. This is because such models must be able to cope
403 with complex and highly variable symptoms and backgrounds, and often must run
404 on low-powered hardware. Growing neural networks take far longer to train/grow
405 than those with fixed topologies but this is of minor concern given efficient par-
406 allelisation and the vast computation resources now available for training. The
407 hardware available to farmers in low income sectors like cocoa, cassava or coffee,
408 however, is restricting. This restriction means that producing a model that is opti-
409 mised for runtime speed at inference is a vital factor and growing neural networks
410 with evolutionary algorithms may be an ideal way to achieve this.
411 Evolving neural networks has been shown to be highly effective in producing neu-
412 ral networks with a high degree of modularity (Amer and Maul, 2019). This in-
413 creased modularity is said to be the result of applying a cost to the number of
414 connections, which both reduces computational cost and promotes evolvability as
415 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 (Schmidt and Bandar, 2001). With evolutionary algorithms, one can also promote diverse populations of networks with techniques like niching (Shir, 2012) and use of non-elitism strategies can allow for the simultaneous exploration of fitness valleys and local optima without getting stuck there (Dang et al., 2021). 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 1 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 dataset, while the top 1 and top 5 accuracy was produced using ImageNet. Fashion MNIST, which is composed of 28x28 pixel grey scale images of clothing (Xiao et al., 2017), 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 (Sun et al., 2020), the number of parameters of VGG16 is miss-reported as 26 million, rather than 138M (*Torchvision Main Documentation* 2023). This sugeststs 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 more simple problem and with

a very low number of parameter when compared with other modern architectures (Table 1 & table 2). However, it does not seem that evolved neural networks are not yet ready to tackle the more difficult problems in plant pathology and so more work is required in this area.

1.6. 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 gives a detailed breakdown of the pros and cons of each of these architectures as well a the number of trainable parameters, which acts as a proxy for model complexity, and the number of giga floating point operations (GFLOPS), which gives a sense of computation cost of running inference with these architectures.

1.6.1. 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 is more computationally demanding than ResNet and ConvNeXT and, while it tends to yield high accuracy on large datasets (Dosovitskiy et al., 2021; Liu et al., 2022), we found that it is prone to overfitting, making it a less favorable 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 (Tan and

Le, 2020). 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 EffifientNet, 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 (Liu et al., 2022). As the first transformer to perform favorably 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 to train on and being less computationally expensive (Dosovitskiy et al., 2021).

1.6.2. Object Detection and Semantic Segmentation Architectures

Although more complex than YOLO, and arguably DETR, Faster-RCNN delivers excellent results and requires only modest resources for training. For most object detection use cases in plant pathology, Faster-RCNN will be the optimal choice. Mask-RCNN extends Faster-RCNN 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 and offers lower precision than Faster-RCNN. 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 (Zhu et al., 2021). However, implementing these architectures can be difficult and they require substantial GPU VRAM for training.

The choice of CV model architecture for a given project depends on a variety of

factors including dataset 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-RCNN will yield the best results while minimising computational cost, risk of overfitting and financial cost of training.

1.7. Image, batch and layer normalisation

In a comparison of a EVAL-COVID (Gong et al., 2021) with other strong competitors like EVOCNN to the detect COVID-19 with evolved CNNs, it was show that the overuse of batch normalisation (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 (Ioffe and Szegedy, 2015), its negative effect on the diagnosis of disease was also observed in case study three, below.

Case study three: Disease detection and normalisation

Here we conducted an ablation study with ResNet18 and ConvNeXt_tiny (Table 3) to asses the effects of image normalisation (IN), batch normalisation (BN) and layer normalisation (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 the purpose of disease detection in all metrics, including training time and overfitting.

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 (Vahdat and Kautz, 2021). As with simply removing the BN layers of a ResNet, reported above, replacing BN in ResNet with the alternative layer normalisation (LN) also results in worse performance (Wu and He, 2018). However, when the authors of ConvNeXt use LN as opposed to BN in their architecture, they observe that the model has no difficulty in training with this substi-

tution (Liu et al., 2022). The BN momentum hyperparameter is a fixed weight applied to the running mean and variance calculations that are tracked during training and used during the application of BN at evaluation or inference time. Thus, adjusting the BN momentum will not affect effect training (Vahdat and Kautz, 2021). However, BN can cause the output of a layer to be slightly shifted during evaluation and a supposed solution to this is to adjust the momentum hyperparameter (Vahdat and Kautz, 2021).

Case study four: Optimisation of BN momentum and image size for cocoa disease detection

While training a cocoa disease detection model we ran a hyperparameter optimisation sweep using the Weights and Biases platform (WANDB) (Biewald, 2023), which included the BN momentum hyperparameter and image input size (Fig. 3). The model architecture used was ResNet18 (He et al., 2016) 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. 100 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 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 above 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 A), 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 B), though this should be optimised for each use case. Previously, image compression has been said to have minor effects on disease detection (Barbedo et al., 2016), while elsewhere it is suggested that image compression should even be avoided completely for small symptoms (Barbedo, 2016) or kept above an arbitrary 1 megapixels (1,000 x 1,000 pixels) (Steddom et al., 2005). However, with the present dataset, which contains images of diseases at varying degrees of progression, using a 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.

Above we have listed a host of reasons why unnecessary normalisation of data is to be avoided. While BN will shorten training time for a CNN, it changes the input data in unpredictable ways, thus worsening prediction results. However, at present the best off-the-shelf CNN that is small enough to run on an older model smart-

phone is ResNet18. So until a more suitable architecture becomes available, BN is unavoidable. We have shown that optimisation of the BN momentum hyperparameter in ResNet18 lead to a slight improvement in the results of our cocoa disease detection model, that IN should not be included in the training pipeline of a model that aims to make predictions from subtle colour features and that excessive image input size should be avoided.

2. DATA ACQUISITION AND MODEL TESTING

In this section we review various interdisciplinary methods available for gathering a training dataset and developing a suitable model. While the previous section was concerned with the theory of ML in CV, this section will focus on practicalities with respect to low cost solutions.

2.1. Obtaining the required training dataset

Training an image classifier to a 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 (Singh et al., 2020). For example, it was found that 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 (Ferentinos, 2018). 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 application. This exemplifies the importance of

curating a realistic, high quality training dataset. By naively training and releasing models that are trained on publicly available datasets, we risk exacerbating the problems of disease miss-classification. At low frequencies, the effect of mislabeled, misleading or uninformative data will have 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 (Motamedi et al., 2021), which act to greatly buffer the effect of infrequent miss-classifications in the training data. However, 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, error resulting from misclassification can be challenging to eradicate. For example, frosty pod rot (FPR), black pod rot (BPR) and witches' broom disease (WBD) in cocoa can all present with black or brown lesions on the pod and both FPR and BPR 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) With careful attention to detail and a detailed knowledge of the pathogen in question, and 2) Using tools and techniques from molecular biology and spectroscopy to better inform model development and subsequent disease detection. Such techniques/tools include DNA sequencing, qPCR, LAMP, MultispeQ and hyperspectral imaging.

2.1.1. Tools from molecular biology

DNA sequencing for the identification of cryptic species (Bickford et al., 2007; Ovaskainen et al., 2010) and plant pathogens (O'Donnell et al., 2015) is now a

common place and invaluable tool. Once sequenced, reads can be used to search previously categorized sequences with the Basic Local Alignment Search Tool (BLAST) from NCBI (Boratyn et al., 2013) to identify a sample by species or other taxonomic group. However, if we know which pathogen we aim to detect, sequencing the whole genome is excessive. Rather, we can use loci like 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 utilised with amplification techniques like polymerase chain reaction (PCR) or Loop-mediated isothermal amplification (LAMP) to detect a pathogen or identify it with relatively low cost and high accuracy. ITS is often used on its own for near-species level identification or in concert with other loci for better specificity (Horton and Bruns, 2001). 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 (Horton and Bruns, 2001).

Quantitative polymerase chain reaction (qPCR) is used to detect asymptomatic disease across the agricultural industry (Luchi et al., 2020). Traditionally PCR has been unsuitable for portable operations or use in the field (Ray et al., 2017). However, rapid real-time PCR in the field is now possible (Schaad and Frederick, 2002). Real-time PCR can also be used to quantify relative levels of a pathogen in plants (Horevaj et al., 2011). 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 for 100 samples) because a thermo cycler is not required, 2) It is fast, 3) Reagents don't 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 the use of this method is that any pre-existing PCR primers cannot be used. This is because PCR primers are designed to amplify a specific region of DNA by binding to complementary sequences on opposite strands of the target DNA. LAMP primers, on the other hand, are designed to 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 detail discussion on the use of ITS amplification in fungal ecology and the potential pitfalls of specific ITS primers design, see Horton and Bruns (2001).

If novel primers are to be designed, the region of interest must first be sequenced and if we aim to identify an as yet unknown pathogen with BLAST, all of the DNA in a sample must be sequenced. Sequencing with the Oxford Nanopore Technology MinION platform can be a an ideal tool for this purpose, offering multiple features: 1) With Oxford Nanopore Technology's field sequencing and library preparation kit, this method allows for sequencing in the field, immediately after tissue samples are gathered. This eliminates the need for cold-chain storage to

avoid sample degradation (ONT, 2023). 2) It allows for high quality sequencing in countries where Illumina sequencing is not available. 3) It is slightly cheaper than using the Illumina platform. 4) The long read length eliminates amplification bias (Goodwin et al., 2015). The avoidance of amplification bias is important for gene expression quantification, which is relevant to the discussion in section 2.2.2. However, the MinION 1B requires a high spec computer and, at £98/sample excluding library preparation, use of this platform also remains too expensive for many projects.

2.1.2. Spectroscopy and hyperspectral imaging

Although not capable of specific disease diagnosis, 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 (Kuhlgert et al., 2022). 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 (Kuhlgert et al., 2022).

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 hyperspectral imaging (HSI). As with the MultispeQ, HSI enables us to detect changes in the chemical composition of biological tissue as conditions such as

675 ripeness or disease status change (Bock et al., 2010). The term 'spectral signature'
676 is used to describe the pattern of electromagnetic radiation reflected by a subject.
677 However, particularly in the case of biology, the term signature is misleading as bi-
678 ological samples often have highly heterogeneous reflectance spectra (Bock et al.,
679 2010). All of the above mentioned CV studies applied ML techniques to RGB im-
680 ages. RGB images capture three discrete bands of the visible spectrum from 400-
681 700 nm. Black and white digital images have two spatial dimensions and a single
682 dimension that describes the darkness of each pixel on a scale of 0-255, whereas
683 RGB images have three colour dimensions represented by values between 0-255,
684 each describing the intensity of red, green or blue light. Hyperspectral images how-
685 ever store a more complete reflectance spectrum for each pixel while also main-
686 taining spatial relationships. The spectral range of these images can be as wide as
687 400-2500 nm (Goetz et al., 1985).
688 Although the applications of hyperspectral photography have long been explored
689 by NASA, this technology is only now becoming cheap enough to be used in in-
690 dustries like agriculture. However, commercially available cameras capable of cap-
691 turing data from the 400-2500 nm range remain expensive and more typically used
692 cameras only sample 400-1000 nm (Table 4). Despite the reduced spectral range of
693 the cheaper cameras, they still provide orders of magnitude more data than RGB
694 cameras, though a lot of this data is highly correlated.
695 Uptake of HSI has recently exploded in a host of fields including archaeology, art
696 conservation, food safety, medicine and crime scene investigation (Lu and Fei,
697 2014). Typical applications of HSI in agriculture include the estimation of yield

(Gutiérrez et al., 2019; Li et al., 2020), assessment of vigour (Feng et al., 2018), remote weed identification (Okamoto et al., 2007), nutrient status (Nguyen et al., 2020) and disease monitoring (Pan et al., 2019).

The analysis of HSI data presents problems which 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 this data and, for a long time, SVMs were the most widely used (Yue et al., 2015). DNNs are now commonly used to analyse this data as they are particularly well suited to the task of classification with HSI data. DNNs have the ability to isolate hidden and complex data structures, they can utilise a great variety of data types, they are flexible in their architectures and the complexity of the functions they can apply, and they are ideally suited to distributed computing (Paoletti et al., 2019). As such, with the addition of dimension reduction techniques such as principal component analysis (Yue et al., 2015), 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 with tangible biological explications of model behaviour.

2.2. Model testing

2.2.1. The black box of DNNs

It is well known how poorly current CV models deal with unexpected edge cases and shifts in test data distribution (Schölkopf et al., 2021). However, in applying CV to plant pathology and agriculture we encounter more cases than most where the test data does not align well with the training data. These problems routinely arise in CV from 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 is contentious how robust of a fix techniques like data augmentations or inductive biases may be to solve the former list of issues (Schölkopf et al., 2021), 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, the role of each dense layer of a CNN has been shown to have distinct roles in feature level extraction and generalisability (Yosinski et al., 2014), and the output of convolution layers have been visualised to show which physical features in an image were more exaggerated (Zeiler and Fergus, 2014). 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 (Toda and Okura, 2019). This work showed that the CNN in question was indeed using visible symptoms of the disease that were similar to those used by human experts. Others have sought to

743 learn how best to actively deceive or manipulate a deep neural network into mis-
744 classification. Working within the remit of cyber security, it was shown that image
745 classifiers based on SVMs and DNNs could easily be deceived with a simple eva-
746 sion algorithm (Biggio et al., 2013). This shows how brittle these classifiers can be
747 and highlights the importance of adopting techniques that rely more heavily on
748 causal inference, such as semi-supervised learning, (Peters et al., 2017) or semantic
749 segmentation. It also highlights the importance of rigorous and conciliatory inter-
750 rogation of models prior to deployment. At present our methods of model evalua-
751 tion are widely considered insufficient and much more work is needed in this area.

752 *2.2.2. Inspecting informative features*

753 A key benefit to the use of CNNs is feature learning. This is the process by which
754 a model will define for itself which features of a dataset it considers informative
755 (Voulodimos et al., 2018). In other CV algorithms, an engineer must handcraft
756 descriptive features of a subject manually, using their expertise and/or diagnostic
757 tools to guide them. In this latter case, pre-processed data are used rather than
758 raw data, as in a CNN. In the convolution layers of a CNN however, kernels and
759 attention weights are applied to raw or augmented image data which emphasise in-
760 formative physical features, and apply inductive biases and self attention, before
761 this data is passed to the dense layer(s) of the network (O'Mahony et al., 2020).
762 We might assume that these physical features would include those that humans
763 consider to be the obvious visible markers for plant disease, such as the presence of
764 lesions on a leaf. However, it is likely that these networks will also identify mark-
765 ers 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 models informative features would tell us if the model is making correct inferences for, what we consider, correct reasons or if it is correct for spurious reasons, suggesting a poor ability to generalise stemming from naive inductive reasoning. Such work may also highlight new ways to identify disease with and without ML or new ways of combating disease spread through phytosanitation, agro-chemistry 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 (Chelebian et al., 2021; Yang and McCord, 2021), the identification of Non-small cell lung cancer subtypes (Yu et al., 2020) and the elucidation of the various functions of drugs (Meyer et al., 2019). CNNs have also been applied alongside transcriptomics in plant science in the investigation of gene regulation in *Arabidopsis* (MacLean, 2019). 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 (Selvaraju et al., 2017; Wang et al., 2018) 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 (Selvaraju et al., 2017). Alternative 'reference based' systems, such as DeepLIFT, rely on back-propagation (Shrikumar et al., 2017) or forward propagation (Explanation map) (Ghosal et al., 2018), using a reference image that does not contain the feature of interest. Applying these methods to miss-classified images can highlight why a model is performing suboptimally (Toda and Okura, 2019) as results produced with these methods have been shown to be highly correlated with assessments of plant disease made by human experts (Ghosal et al., 2018).

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. The 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 (Howard et al., 2019) will help to keep computational cost down and, after training, methods such as pruning and quantisation may reduce this cost further. While Google Colab offers free limited access to GPUs for model training, the rental cost of a 16GB

NVIDIA V100 GPU, which would be the minimum needed to train a transformer model or large CNN, is $2.48/hour. As such, developing and training such large models for days, or even weeks, can soon become expensive.

ONNX Runtime from Microsoft (Microsoft, 2023) offers a huge array of tools to help accelerate, quantise and deploy trained DNNs. Such models can be incorporated into Android or IOS apps, using the phones builtin camera, they can be deployed via the web, on edge devices like 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 GELU and stochastic depth, may cause problems as these operators are not yet supported. TensorFlow also offers a pipeline for model deployment and the Pytorch toolkit for techniques like quantisation aware training and model compression is maturing, but presented difficulties when we attempted to use it. In contrast, the ONNX Runtime pipeline is extremely easy to use and supports all popular model formats like Pytorch, TensorFlow and SciKit Learn. 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 (Solodskikh et al., 2023), 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 implementation of object detection or segmentation models, we recommend the Detectron2 library from Facebook (facebookresearch, 2023). This library incorporates Faster-RCNN, Mask-RCNN and some new transformers models like ViTDet,

834 and offers a host of tutorials on the whole process from training to implementa-
835 tion.

836 Conclusion

837 Described here are all of the tools necessary to develop highly optimised and ro-
838 bust ML models that use minimal computational power and provide real benefit to
839 sectors that have more modest budgets. The application of these tools will allow
840 us to break from the common trend in the ML industry, where expensive hard-
841 ware is employed to develop complex and computationally expensive models to the
842 detriment of improving training data quality.
843 With the application of off-the-shelf architectures to stock datasets, such as the
844 plant village dataset (Geetharamani and Pandian, 2019), we can easily achieve pre-
845 diction accuracy scores in the high 90% range (Thapa et al., 2020). However, such
846 models have little value because they will not generalise to complex real-world en-
847 vironments due to the simplicity of the training data.
848 We offer the following recommendations for the development of efficient, inexpen-
849 sive and robust CV models for plant pathology.
850 **Garbage in - garbage out:** The thoughtless application of advanced models to
851 poorly labeled, simplistic, contaminated or maltransformed data will yield models
852 that have little value in the field, with slow inference time, poor accuracy and an
853 inability to generalise. To avoid this fate we should; (A) where possible, consult
854 with specialists and utilise the invaluable tools from biology, chemistry and spec-
855 troscopy to label data, (B) use the minimum appropriate image input size to im-

prove runtime speed and help avoid overfitting, and (C) avoid needless data transformations like normalisation, which can alter data in unreliable ways.

The potential in training procedures: Techniques like semantic segmentation and semi-supervised learning have potential to lessen both bias and variance in a models 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 modeling methods, such as SVMs, still have a role to play in modern computer vision, 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 vital therefore that we test the models we develop rigorously to ensure that they are not prone to miss-classification born of overfitting and naive generalisations. While metrics such as accuracy, F1, 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 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 2.2.2.

If we apply our wealth of knowledge and proven techniques from botany and agron-

omy 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 (UN, 2022). 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, such as semi-supervised learning, evolving neural networks and incorporation of omics to model development may enable us, as a community of growers, botanists and ML 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.

AUTHOR CONTRIBUTIONS

J.R.S. conceived of this review, read and summarized the literature, and wrote the first draft of the manuscript. K.J.D. and D.W.F. continually reviewed and edited the manuscript and approved the final manuscript before submission and publication.

Acknowledgments

This work was made possible by funding from the Doctoral Centre for Safe, Ethical and Secure Computing at the University of York.

Data availability

Case study one: Semantic segmentation for cocoa disease detection The image data, annotations and link to the accompanying Github repository can be found at:

`osf.io/79kx3/?view_only=4a2c1dccee1a4baeb85de5002c702f10`

Case study two: Semi-supervised learning for outlier detection The data to train the initial supervised model, the .csv search terms file for the below web scraper and the final semi-supervised model weights can be found at: `osf.io/h5gj7/?view_only=dbf9f245e21a41e185f5b73e718b4cad` The 'contaminated' data used to train the semi-supervised model was generated using the code at: `github.com/jrsykes/Google-Image-Scraper`

The custom code used to train both the initial model and the final semi-supervised model can be found at:

`github.com/jrsykes/CocoaReader/blob/main/PlantNotPlant`

Case study three: Disease detection and normalisation The custom code used to conduct this study can be found in the following Github reposetory, with acompanying Readme.md: `github.com/jrsykes/CocoaReader`

The data for this study was scraped from the internet using the code in to following github repository: `github.com/jrsykes/Google-Image-Scraper`

The location of the accompanying ".csv search terms file" is described below.

Case study four: Optimisation of BN momentum and image size The custom code to run this sweep can be found at the following Github reposetory: `github.`

`com/jrsykes/CocoaReader/tree/main/CocoaNet`

The main script is titled CocoNetsweep_min.sh and the wandb config file is titled CocoaNetSweepConfig_min.yml. The data used to generate these results and the full wandb report can be found at:

`osf.io/2fw6g/?view_only=adc66ba66f83465a9e7b111515a60bf2`

REFERENCES

Amer, M. and T. Maul, (2019). "A Review of Modularization Techniques in Artificial Neural Networks". In: *Artificial Intelligence Review* 52.1, pp. 527–561.

An, J. and S. Cho, (2015). "Variational Autoencoder Based Anomaly Detection Using Reconstruction Probability". In: *Special lecture on IE*.

Asperti, A., D. Evangelista, and E. Loli Piccolomini, (2021). "A Survey on Variational Autoencoders from a Green AI Perspective". In: *SN Computer Science* 2.4, p. 301.

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

Barbedo, J. G. A., (2016). "A Review on the Main Challenges in Automatic Plant Disease Identification Based on Visible Range Images". In: *Biosystems Engineering* 144, pp. 52–60.

941 Barbedo, J. G. A., L. V. Koenigkan, and T. T. Santos, (2016). "Identifying Multi-
942 ple Plant Diseases Using Digital Image Processing". In: *Biosystems Engineering*
943 147, pp. 104–116.
944 Bharati, P. and A. Pramanik, (2020). "Deep Learning Techniques—R-CNN to
945 Mask R-CNN: A Survey". In: *Computational Intelligence in Pattern Recogni-*
946 *tion*. Ed. by A. K. Das, J. Nayak, B. Naik, S. K. Pati, and D. Pelusi. Advances
947 in Intelligent Systems and Computing. Singapore: Springer, pp. 657–668.
948 Bickford, D., D. J. Lohman, N. S. Sodhi, P. K. L. Ng, R. Meier, K. Winker, K. K.
949 Ingram, and I. Das, (2007). "Cryptic Species as a Window on Diversity and
950 Conservation". In: *Trends in Ecology & Evolution* 22.3, pp. 148–155.
951 Biewald, L., (2023). *Experiment Tracking with Weights and Biases*.
952 Biggio, B., I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto,
953 and F. Roli, (2013). "Evasion Attacks against Machine Learning at Test Time".
954 In: *Advanced Information Systems Engineering*. Ed. by C. Salinesi, M. C. Nor-
955 rie, and Ó. Pastor. Red. by D. Hutchison, T. Kanade, J. Kittler, J. M. Klein-
956 berg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan,
957 B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, and G. Weikum.
958 Vol. 7908. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 387–402.
959 Bock, C. H., G. H. Poole, P. E. Parker, and T. R. Gottwald, (2010). "Plant Dis-
960 ease Severity Estimated Visually, by Digital Photography and Image Analysis,
961 and by Hyperspectral Imaging". In: *Critical Reviews in Plant Sciences* 29.2,
962 pp. 59–107.

963 Boratyn, G. M., C. Camacho, P. S. Cooper, G. Coulouris, A. Fong, N. Ma, T. L.
964 Madden, W. T. Matten, S. D. McGinnis, Y. Merezhuk, Y. Raytselis, E. W.
965 Sayers, T. Tao, J. Ye, and I. Zaretskaya, (2013). "BLAST: A More Efficient Re-
966 port with Usability Improvements". In: *Nucleic Acids Research* 41.W1, W29–
967 W33.
968 Cao, S. and R. Nevatia, (2016). "Exploring Deep Learning Based Solutions in Fine
969 Grained Activity Recognition in the Wild". In: *2016 23rd International Con-*
970 *ference on Pattern Recognition (ICPR)*. 2016 23rd International Conference on
971 Pattern Recognition (ICPR), pp. 384–389.
972 Carion, N., F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko,
973 (2020). "End-to-End Object Detection with Transformers". In: *Computer Vi-*
974 *sion – ECCV 2020*. Ed. by A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm.
975 Lecture Notes in Computer Science. Cham: Springer International Publishing,
976 pp. 213–229.
977 Chelebian, E., C. Avenel, K. Kartasalo, M. Marklund, A. Tanoglidi, T. Mirtti, R.
978 Colling, A. Erickson, A. D. Lamb, J. Lundeberg, and C. Wählby, (2021). "Mor-
979 phological Features Extracted by AI Associated with Spatial Transcriptomics in
980 Prostate Cancer". In: *Cancers* 13.19 (19), p. 4837.
981 Chiu, M. T., X. Xu, Y. Wei, Z. Huang, A. G. Schwing, R. Brunner, H. Khacha-
982 trian, H. Karapetyan, I. Dozier, G. Rose, D. Wilson, A. Tudor, N. Hovakimyan,
983 T. S. Huang, and H. Shi, (2020). "Agriculture-Vision: A Large Aerial Image
984 Database for Agricultural Pattern Analysis". In: Proceedings of the IEEE/CVF
985 Conference on Computer Vision and Pattern Recognition, pp. 2828–2838.

Chouhan, S. S., U. P. Singh, and S. Jain, (2020). "Applications of Computer Vision in Plant Pathology: A Survey". In: *Archives of Computational Methods in Engineering* 27.2, pp. 611–632.

Cohen, Y. and M. D. Coffey, (1986). "Systemic Fungicides and the Control of Oomycetes". In: *Annual Review of Phytopathology* 24.1, pp. 311–338.

Dang, D.-C., A. Eremeev, and P. K. Lehre, (2021). "Escaping Local Optima with Non-Elitist Evolutionary Algorithms". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 35.14 (14), pp. 12275–12283.

Department of Health. Victoria, A., (2023). *Methyl Bromide Use in Victoria Community Factsheet*. URL: `https://www.health.vic.gov.au/publications/methyl-bromide-use-in-victoria-community-factsheet` (visited on 06/26/2023).

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

Dosovitskiy, A., L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, (2021). *An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale*. URL: `http://arxiv.org/abs/2010.11929` (visited on 11/14/2022). preprint.

facebookresearch, (2023). *Facebookresearch*. Meta Research.

Fan, Y., G. Wen, D. Li, S. Qiu, M. D. Levine, and F. Xiao, (2020). "Video Anomaly Detection and Localization via Gaussian Mixture Fully Convolutional Vari-

ational Autoencoder". In: *Computer Vision and Image Understanding* 195, p. 102920.

Feng, L., S. Zhu, C. Zhang, Y. Bao, X. Feng, and Y. He, (2018). "Identification of Maize Kernel Vigor under Different Accelerated Aging Times Using Hyperspectral Imaging". In: *Molecules* 23.12 (12), p. 3078.

Ferentinos, K. P., (2018). "Deep Learning Models for Plant Disease Detection and Diagnosis". In: *Computers and Electronics in Agriculture* 145, pp. 311–318.

Fuentes, S., G. Chacon, D. D. Torrico, A. Zarate, and C. Gonzalez Viejo, (2019). "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". In: *Sensors* 19.14 (14), p. 3054.

Geetharamani, G. and A. Pandian, (2019). "Identification of Plant Leaf Diseases Using a Nine-Layer Deep Convolutional Neural Network". In: *Computers & Electrical Engineering* 76, pp. 323–338.

Ghosal, S., D. Blystone, A. K. Singh, B. Ganapathysubramanian, A. Singh, and S. Sarkar, (2018). "An Explainable Deep Machine Vision Framework for Plant Stress Phenotyping". In: *Proceedings of the National Academy of Sciences* 115.18, pp. 4613–4618.

Goetz, A. F. H., G. Vane, J. E. Solomon, and B. N. Rock, (1985). "Imaging Spectrometry for Earth Remote Sensing". In: *Science* 228.4704, pp. 1147–1153.

Gong, Y., Y. Sun, D. Peng, P. Chen, Z. Yan, and K. Yang, (2021). "Analyze COVID-19 CT Images Based on Evolutionary Algorithm with Dynamic Searching Space". In: *Complex & Intelligent Systems* 7.6, pp. 3195–3209.

1032 Goodwin, S., J. Gurtowski, S. Ethe-Sayers, P. Deshpande, M. C. Schatz, and W. R.
1033 McCombie, (2015). "Oxford Nanopore Sequencing, Hybrid Error Correction,
1034 and de Novo Assembly of a Eukaryotic Genome". In: *Genome Research* 25.11,
1035 pp. 1750–1756.
1036 Grosch, K., (2018). *John Deere – Bringing AI to Agriculture*. Technology and Op-
1037 erations Management. URL: `https://digital.hbs.edu/platform-rctom/submission/john-deere-bringing-ai-to-agriculture/` (visited on
1039 05/16/2022).
1040 Guo, M.-H., T.-X. Xu, J.-J. Liu, Z.-N. Liu, P.-T. Jiang, T.-J. Mu, S.-H. Zhang,
1041 R. R. Martin, M.-M. Cheng, and S.-M. Hu, (2022). "Attention Mechanisms in
1042 Computer Vision: A Survey". In: *Computational Visual Media* 8.3, pp. 331–368.
1043 Gutiérrez, S., A. Wendel, and J. Underwood, (2019). "Ground Based Hyperspec-
1044 tral Imaging for Extensive Mango Yield Estimation". In: *Computers and Elec-*
1045 *tronics in Agriculture* 157, pp. 126–135.
1046 He, K., G. Gkioxari, P. Dollar, and R. Girshick, (2017). "Mask R-CNN". In: Pro-
1047 ceedings of the IEEE International Conference on Computer Vision, pp. 2961–
1048 2969.
1049 He, K., X. Zhang, S. Ren, and J. Sun, (2016). "Deep Residual Learning for Image
1050 Recognition". In: Proceedings of the IEEE Conference on Computer Vision and
1051 Pattern Recognition, pp. 770–778.
1052 Horevaj, P., E. Milus, and B. Bluhm, (2011). "A Real-Time qPCR Assay to Quan-
1053 tify Fusarium Graminearum Biomass in Wheat Kernels". In: *Journal of Applied*
1054 *Microbiology* 111.2, pp. 396–406.

1055 Horton, T. R. and T. D. Bruns, (2001). "The Molecular Revolution in Ectomy-
1056 corrhizal Ecology: Peeking into the Black-Box". In: *Molecular Ecology* 10.8,
1057 pp. 1855–1871.
1058 Horzyk, A. and E. Ergün, (2020). "YOLOv3 Precision Improvement by the Weighted
1059 Centers of Confidence Selection". In: *2020 International Joint Conference on*
1060 *Neural Networks (IJCNN)*. 2020 International Joint Conference on Neural Net-
1061 works (IJCNN), pp. 1–8.
1062 Howard, A., M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu,
1063 R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, (2019). *Searching for Mo-*
1064 *bileNetV3*. URL: `http://arxiv.org/abs/1905.02244` (visited on 06/21/2023).
1065 preprint.
1066 Howard, A. G., M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. An-
1067 dreetto, and H. Adam, (2017). *MobileNets: Efficient Convolutional Neural Net-*
1068 *works for Mobile Vision Applications*. URL: `http://arxiv.org/abs/1704.`
1069 `04861` (visited on 10/14/2022). preprint.
1070 Huang, H., Z. Wei, and L. Yao, (2019). "A Novel Approach to Component Assem-
1071 bly Inspection Based on Mask R-CNN and Support Vector Machines". In: *In-*
1072 *formation* 10.9 (9), p. 282.
1073 Huda-Shakirah, A. R., N. M. I. Mohamed Nor, L. Zakaria, Y.-H. Leong, and M. H.
1074 Mohd, (2022). "Lasiodiplodia Theobromae as a Causal Pathogen of Leaf Blight,
1075 Stem Canker, and Pod Rot of Theobroma Cacao in Malaysia". In: *Scientific*
1076 *Reports* 12.1 (1), p. 8966.

1077 Ioffe, S. and C. Szegedy, (2015). "Batch Normalization: Accelerating Deep Network
1078 Training by Reducing Internal Covariate Shift". In: *Proceedings of the 32nd In-*
1079 *ternational Conference on Machine Learning*. International Conference on Ma-
1080 chine Learning. PMLR, pp. 448–456.
1081 Khanzada, M., A. Lodhi, and S. Shahzad, (2005). "Chemical Control of Lasiodiplo-
1082 dia Theobromae, the Causal Agent of Mango Decline in Sindh". In: *Pakistan*
1083 *Journal of Botany* 37, pp. 1023–1030.
1084 Kingma, D. P. and M. Welling, (2019). "An Introduction to Variational Autoen-
1085 coders". In: *Foundations and Trends® in Machine Learning* 12.4, pp. 307–392.
1086 Krizhevsky, A., I. Sutskever, and G. E. Hinton, (2017). "ImageNet Classification
1087 with Deep Convolutional Neural Networks". In: *Communications of the ACM*
1088 60.6, pp. 84–90.
1089 Kroon, L. P. N. M., H. Brouwer, A. W. A. M. de Cock, and F. Govers, (2012).
1090 "The Genus Phytophthora Anno 2012". In: *Phytopathology®* 102.4, pp. 348–
1091 364.
1092 Kuhlgert, S., G. Austic, R. Zegarac, I. Osei-Bonsu, D. Hoh, M. I. Chilvers, M. G.
1093 Roth, K. Bi, D. TerAvest, P. Weebadde, and D. M. Kramer, (2022). "Multi-
1094 speQ Beta: A Tool for Large-Scale Plant Phenotyping Connected to the Open
1095 PhotosynQ Network". In: *Royal Society Open Science* 3.10 (), p. 160592.
1096 Kuok Ho, D. T. and P. S. Yap, (2020). *A Systematic Review of Slash-and-Burn*
1097 *Agriculture as an Obstacle to Future-Proofing Climate Change*, p. 19. 01 p.
1098 Li, B., X. Xu, L. Zhang, J. Han, C. Bian, G. Li, J. Liu, and L. Jin, (2020). "Above-
1099 Ground Biomass Estimation and Yield Prediction in Potato by Using UAV-

based RGB and Hyperspectral Imaging". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 162, pp. 161–172.

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

Li, X. and J. She, (2017). "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: Association for Computing Machinery, pp. 305–314.

Lim, K.-L., X. Jiang, and C. Yi, (2020). "Deep Clustering With Variational Autoencoder". In: *IEEE Signal Processing Letters* 27, pp. 231–235.

Lin, E., S. Mukherjee, and S. Kannan, (2020). "A Deep Adversarial Variational Autoencoder Model for Dimensionality Reduction in Single-Cell RNA Sequencing Analysis". In: *BMC Bioinformatics* 21.1, p. 64.

Liu, Z., H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, (2022). "A ConvNet for the 2020s".

Liu, Z., P. Luo, X. Wang, and X. Tang, (2015). "Deep Learning Face Attributes in the Wild". In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3730–3738.

Lopes, J. F., V. G. T. da Costa, D. F. Barbin, L. J. P. Cruz-Tirado, V. Baeten, and S. Barbon Junior, (2022). "Deep Computer Vision System for Cocoa Classification". In: *Multimedia Tools and Applications* 81.28, pp. 41059–41077.

1123 Lu, G. and B. Fei, (2014). "Medical Hyperspectral Imaging: A Review". In: *Jour-*
1124 *nal of Biomedical Optics* 19.1, p. 010901.
1125 Luchi, N., R. Ioos, and A. Santini, (2020). "Fast and Reliable Molecular Methods
1126 to Detect Fungal Pathogens in Woody Plants". In: *Applied Microbiology and*
1127 *Biotechnology* 104.6, pp. 2453–2468.
1128 Maaløe, L., M. Fraccaro, V. Liévin, and O. Winther, (2019). "BIVA: A Very Deep
1129 Hierarchy of Latent Variables for Generative Modeling". In: *Advances in Neural*
1130 *Information Processing Systems*. Vol. 32. Curran Associates, Inc.
1131 Mack, C. A., (2011). "Fifty Years of Moore's Law". In: *IEEE Transactions on*
1132 *Semiconductor Manufacturing* 24.2, pp. 202–207.
1133 MacLean, D., (2019). "A Convolutional Neural Network for Predicting Transcrip-
1134 tional Regulators of Genes in Arabidopsis Transcriptome Data Reveals Classifi-
1135 cation Based on Positive Regulatory Interactions". In: p. 618926.
1136 Maddison, A. C., G. Macias, C. Moreira, R. Arias, and R. Neira, (1995). "Cocoa
1137 Production in Ecuador in Relation to Dry-Season Escape from Pod Rot Caused
1138 by Crinipellis Perniciosa and Moniliophthora Roreri". In: *Plant Pathology* 44.6,
1139 pp. 982–998.
1140 Malhi, Y., J. T. Roberts, R. A. Betts, T. J. Killeen, W. Li, and C. A. Nobre, (2008).
1141 "Climate Change, Deforestation, and the Fate of the Amazon". In: *Science*
1142 319.5860, pp. 169–172.
1143 Marelli, J.-P., D. I. Guest, B. A. Bailey, H. C. Evans, J. K. Brown, M. Junaid,
1144 R. W. Barreto, D. O. Lisboa, and A. S. Puig, (2019). "Chocolate Under Threat
1145 from Old and New Cacao Diseases". In: *Phytopathology®* 109.8, pp. 1331–1343.

1146 Meinhardt, L. W., J. Rincones, B. A. Bailey, M. C. Aime, G. W. Griffith, D. Zhang,
1147 and G. a. G. Pereira, (2008). "Moniliophthora Perniciosa, the Causal Agent of
1148 Witches' Broom Disease of Cacao: What's New from This Old Foe?" In: *Molec-*
1149 *ular Plant Pathology* 9.5, pp. 577–588.
1150 Meyer, J. G., S. Liu, I. J. Miller, J. J. Coon, and A. Gitter, (2019). "Learning
1151 Drug Functions from Chemical Structures with Convolutional Neural Networks
1152 and Random Forests". In: *Journal of Chemical Information and Modeling* 59.10,
1153 pp. 4438–4449.
1154 Microsoft, (2023). *ONNX Runtime*. URL: `https://onnxruntime.ai/` (visited on
1155 06/21/2023).
1156 Mite-Baidal, K., E. Solís-Avilés, T. Martínez-Carriel, A. Marcillo-Plaza, E. Cruz-
1157 Ibarra, and W. Baque-Bustamante, (2019). "Analysis of Computer Vision Al-
1158 gorithms to Determine the Quality of Fermented Cocoa (Theobroma Cacao):
1159 Systematic Literature Review". In: *ICT for Agriculture and Environment*. Ed.
1160 by R. Valencia-García, G. Alcaraz-Mármol, J. del Cioppo-Morstadt, N. Vera-
1161 Lucio, and M. Bucaram-Leverone. Advances in Intelligent Systems and Com-
1162 puting. Cham: Springer International Publishing, pp. 79–87.
1163 Motamedi, M., N. Sakharnykh, and T. Kaldewey, (2021). "A Data-Centric Ap-
1164 proach for Training Deep Neural Networks with Less Data".
1165 Nagarajan, S., G. Seibold, J. Kranza, E. E. Saari, and L. M. Joshi, (1984). "Moni-
1166 toring Wheat Rust Epidemics With the Landsat-2 Satellite". In: *Phytopathology*
1167 74.5, p. 585.

1168 Nalisnick, E., A. Matsukawa, Y. W. Teh, D. Gorur, and B. Lakshminarayanan,
1169 (2019). *Do Deep Generative Models Know What They Don't Know?* URL: `http:`
1170 `//arxiv.org/abs/1810.09136` (visited on 11/10/2022). preprint.
1171 Nguyen, H. D. D., V. Pan, C. Pham, R. Valdez, K. Doan, and C. Nansen, (2020).
1172 "Night-Based Hyperspectral Imaging to Study Association of Horticultural
1173 Crop Leaf Reflectance and Nutrient Status". In: *Computers and Electronics*
1174 *in Agriculture* 173, p. 105458.
1175 O'Donnell, K., T. J. Ward, V. A. R. G. Robert, P. W. Crous, D. M. Geiser, and
1176 S. Kang, (2015). "DNA Sequence-Based Identification of Fusarium: Current
1177 Status and Future Directions". In: *Phytoparasitica* 43.5, pp. 583–595.
1178 O'Mahony, N., S. Campbell, A. Carvalho, S. Harapanahalli, G. V. Hernandez, L.
1179 Krpalkova, D. Riordan, and J. Walsh, (2020). "Deep Learning vs. Traditional
1180 Computer Vision". In: *Advances in Computer Vision*. Ed. by K. Arai and S.
1181 Kapoor. Advances in Intelligent Systems and Computing. Cham: Springer In-
1182 ternational Publishing, pp. 128–144.
1183 Okamoto, H., T. Murata, T. Kataoka, and S.-I. Hata, (2007). "Plant Classification
1184 for Weed Detection Using Hyperspectral Imaging with Wavelet Analysis". In:
1185 *Weed Biology and Management* 7.1, pp. 31–37.
1186 Oliveira, M. M., B. V. Cerqueira, S. Barbon, and D. F. Barbin, (2021). "Classi-
1187 fication of Fermented Cocoa Beans (Cut Test) Using Computer Vision". In:
1188 *Journal of Food Composition and Analysis* 97, p. 103771.
1189 ONT, (2023). *Field Sequencing Kit*. URL: `https://store.nanoporetech.com/`
1190 `field-sequencing-kit.html` (visited on 02/01/2023).

1191 Ovaskainen, O., J. Nokso-Koivisto, J. Hottola, T. Rajala, T. Pennanen, H. Ali-
1192 Kovero, O. Miettinen, P. Oinonen, P. Auvinen, L. Paulin, K.-H. Larsson, and
1193 R. Mäkipää, (2010). "Identifying Wood-Inhabiting Fungi with 454 Sequencing
1194 – What Is the Probability That BLAST Gives the Correct Species?" In: *Fungal*
1195 *Ecology* 3.4, pp. 274–283.
1196 Pan, T.-t., E. Chyngyz, D.-W. Sun, J. Paliwal, and H. Pu, (2019). "Pathogenetic
1197 Process Monitoring and Early Detection of Pear Black Spot Disease Caused by
1198 Alternaria Alternata Using Hyperspectral Imaging". In: *Postharvest Biology*
1199 *and Technology* 154, pp. 96–104.
1200 Paoletti, M. E., J. M. Haut, J. Plaza, and A. Plaza, (2019). "Deep Learning Clas-
1201 sifiers for Hyperspectral Imaging: A Review". In: *ISPRS Journal of Photogram-*
1202 *metry and Remote Sensing* 158, pp. 279–317.
1203 Parra, P., T. Negrete, J. Llaguno, and N. Vega, (2018). "Computer Vision Tech-
1204 niques Applied in the Estimation of the Cocoa Beans Fermentation Grade". In:
1205 *2018 IEEE ANDESCON*. 2018 IEEE ANDESCON, pp. 1–10.
1206 Patrício, D. I. and R. Rieder, (2018). "Computer Vision and Artificial Intelligence
1207 in Precision Agriculture for Grain Crops: A Systematic Review". In: *Computers*
1208 *and Electronics in Agriculture* 153, pp. 69–81.
1209 Peters, J., D. Janzing, and B. Schölkopf, (2017). *Elements of Causal Inference:*
1210 *Foundations and Learning Algorithms*. The MIT Press.
1211 Phillips-Mora, W. and M. J. Wilkinson, (2007). "Frosty Pod of Cacao: A Disease
1212 with a Limited Geographic Range but Unlimited Potential for Damage". In:
1213 *Phytopathology®* 97.12, pp. 1644–1647.

Ramos-Giraldo, P., C. Reberg-Horton, A. M. Locke, S. Mirsky, and E. Lobaton, (2020). "Drought Stress Detection Using Low-Cost Computer Vision Systems and Machine Learning Techniques". In: *IT Professional* 22.3, pp. 27–29.

Ray, M., A. Ray, S. Dash, A. Mishra, K. G. Achary, S. Nayak, and S. Singh, (2017). "Fungal Disease Detection in Plants: Traditional Assays, Novel Diagnostic Techniques and Biosensors". In: *Biosensors and Bioelectronics* 87, pp. 708–723.

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

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

Rice, R. A. and R. Greenberg, (2000). "Cacao Cultivation and the Conservation of Biological Diversity". In: *AMBIO: A Journal of the Human Environment* 29.3, pp. 167–173.

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

Salvatore, M. M., A. Andolfi, and R. Nicoletti, (2020). "The Thin Line between Pathogenicity and Endophytism: The Case of Lasiodiplodia Theobromae". In: *Agriculture* 10.10 (10), p. 488.

Sarkate, R. S., N. V. Kalyankar, and P. B. Khanale, (2013). "Application of Computer Vision and Color Image Segmentation for Yield Prediction Precision".

In: *2013 International Conference on Information Systems and Computer Networks*. 2013 International Conference on Information Systems and Computer Networks, pp. 9–13.

Schaad, N. W. and R. D. Frederick, (2002). "Real-Time PCR and Its Application for Rapid Plant Disease Diagnostics". In: *Canadian Journal of Plant Pathology* 24.3, pp. 250–258.

Schmidt, A. and Z. Bandar, (2001). "Modularity - A Concept For New Neural Network Architectures". In.

Schölkopf, B., F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio, (2021). "Toward Causal Representation Learning". In: *Proceedings of the IEEE* 109.5, pp. 612–634.

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

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

Shrikumar, A., P. Greenside, and A. Kundaje, (2017). "Learning Important Features Through Propagating Activation Differences". In: *Proceedings of the 34th International Conference on Machine Learning*. International Conference on Machine Learning. PMLR, pp. 3145–3153.

Singh, D., N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, (2020). "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: Association for Computing Machinery, pp. 249–253.

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

Statista, (2022). *Smartphone Users 2026*. Statista. URL: `https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/` (visited on 05/16/2022).

Steddom, K., M. McMullen, B. Schatz, and C. M. Rush, (2005). "Comparing Image Format and Resolution for Assessment of Foliar Diseases of Wheat". In: *Plant Health Progress* 6.1, p. 11.

Su, J., C. Liu, M. Coombes, X. Hu, C. Wang, X. Xu, Q. Li, L. Guo, and W.-H. Chen, (2018). "Wheat Yellow Rust Monitoring by Learning from Multispectral UAV Aerial Imagery". In: *Computers and Electronics in Agriculture* 155, pp. 157–166.

Sun, J., X. Wang, N. Xiong, and J. Shao, (2018). "Learning Sparse Representation With Variational Auto-Encoder for Anomaly Detection". In: *IEEE Access* 6, pp. 33353–33361.

Sun, Y., B. Xue, M. Zhang, and G. G. Yen, (2020). "Evolving Deep Convolutional Neural Networks for Image Classification". In: *IEEE Transactions on Evolutionary Computation* 24.2, pp. 394–407.

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

Tan, M. and Q. V. Le, (2020). *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*. URL: `http://arxiv.org/abs/1905.11946` (visited on 06/21/2023). preprint.

Thapa, R., K. Zhang, N. Snavely, S. Belongie, and A. Khan, (2020). "The Plant Pathology Challenge 2020 Data Set to Classify Foliar Disease of Apples". In: *Applications in Plant Sciences* 8.9, e11390.

Tian, H., T. Wang, Y. Liu, X. Qiao, and Y. Li, (2020). "Computer Vision Technology in Agricultural Automation —A Review". In: *Information Processing in Agriculture* 7.1, pp. 1–19.

Toda, Y. and F. Okura, (2019). "How Convolutional Neural Networks Diagnose Plant Disease". In: *Plant Phenomics* 2019.

Torchvision Main Documentation (2023). URL: `https://pytorch.org/vision/main/models` (visited on 06/21/2023).

Tripathi, M. K. and D. D. Maktedar, (2020). "A Role of Computer Vision in Fruits and Vegetables among Various Horticulture Products of Agriculture Fields: A Survey". In: *Information Processing in Agriculture* 7.2, pp. 183–203.

1303 UN, (2022). *United Nations*. World Population Prospects 2022. URL: https://
1304 population.un.org/wpp/Graphs/Probabilistic/POP/TOT/900 (visited on
1305 05/18/2022).
1306 Vahdat, A. and J. Kautz, (2021). *NVAE: A Deep Hierarchical Variational Autoen-*
1307 *coder*. URL: http://arxiv.org/abs/2007.03898 (visited on 07/12/2022).
1308 preprint.
1309 Voulodimos, A., N. Doulamis, A. Doulamis, and E. Protopapadakis, (2018). "Deep
1310 Learning for Computer Vision: A Brief Review". In: *Computational Intelligence*
1311 *and Neuroscience* 2018, e7068349.
1312 Vuola, A. O., S. U. Akram, and J. Kannala, (2019). "Mask-RCNN and U-Net En-
1313 sembled for Nuclei Segmentation". In: *2019 IEEE 16th International Sympo-*
1314 *sium on Biomedical Imaging (ISBI 2019)*. 2019 IEEE 16th International Sym-
1315 posium on Biomedical Imaging (ISBI 2019), pp. 208–212.
1316 Wang, L., Z. Wu, S. Karanam, K.-C. Peng, and R. Vikram Singh, (2018). *Reducing*
1317 *Visual Confusion with Discriminative Attention*.
1318 Weinstein, B. G., (2018). "A Computer Vision for Animal Ecology". In: *Journal of*
1319 *Animal Ecology* 87.3, pp. 533–545.
1320 Wu, Y. and K. He, (2018). "Group Normalization". In: p. 17.
1321 Wu, Z., Y. Chen, B. Zhao, X. Kang, and Y. Ding, (2021). "Review of Weed Detec-
1322 tion Methods Based on Computer Vision". In: *Sensors* 21.11 (11), p. 3647.
1323 Xiao, H., K. Rasul, and R. Vollgraf, (2017). *Fashion-MNIST: A Novel Image Dataset*
1324 *for Benchmarking Machine Learning Algorithms*. URL: http://arxiv.org/
1325 abs/1708.07747 (visited on 06/23/2023). preprint.

1326 Xie, S., R. Girshick, P. Dollár, Z. Tu, and K. He, (2017). *Aggregated Residual Trans-*
1327 *formations for Deep Neural Networks*. URL: `http://arxiv.org/abs/1611.`
1328 `05431` (visited on 06/23/2023). preprint.
1329 Xu, S., J. Wang, W. Shou, T. Ngo, A.-M. Sadick, and X. Wang, (2021). "Com-
1330 puter Vision Techniques in Construction: A Critical Review". In: *Archives of*
1331 *Computational Methods in Engineering* 28.5, pp. 3383–3397.
1332 Xu, W., H. Sun, C. Deng, and Y. Tan, (2017). "Variational Autoencoder for Semi-
1333 Supervised Text Classification". In: *Thirty-First AAAI Conference on Artificial*
1334 *Intelligence*. Thirty-First AAAI Conference on Artificial Intelligence.
1335 Xu, W. and Y. Tan, (2020). "Semisupervised Text Classification by Variational
1336 Autoencoder". In: *IEEE Transactions on Neural Networks and Learning Sys-*
1337 *tems* 31.1, pp. 295–308.
1338 Yang, X. and R. P. McCord, (2021). "CoSTA: Unsupervised Convolutional Neural
1339 Network Learning for Spatial Transcriptomics Analysis". In: *bioRxiv*, p. 26.
1340 Yosinski, J., J. Clune, Y. Bengio, and H. Lipson, (2014). "How Transferable Are
1341 Features in Deep Neural Networks?" In: *Advances in Neural Information Pro-*
1342 *cessing Systems*. Vol. 27. Curran Associates, Inc.
1343 Yu, K.-H., F. Wang, G. J. Berry, C. Ré, R. B. Altman, M. Snyder, and I. S. Ko-
1344 hane, (2020). "Classifying Non-Small Cell Lung Cancer Types and Transcrip-
1345 tomic Subtypes Using Convolutional Neural Networks". In: *Journal of the Amer-*
1346 *ican Medical Informatics Association* 27.5, pp. 757–769.

Yue, J., W. Zhao, S. Mao, and H. Liu, (2015). "Spectral–Spatial Classification of Hyperspectral Images Using Deep Convolutional Neural Networks". In: *Remote Sensing Letters* 6.6, pp. 468–477.

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

Zhang, X., X. Zhou, M. Lin, and J. Sun, (2018). "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.

Zhao, T., Y. Yang, H. Niu, D. Wang, and Y. Chen, (2018). "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*. Multispectral, Hyperspectral, and Ultraspectral Remote Sensing Technology, Techniques and Applications VII. Vol. 10780. SPIE, pp. 210–218.

Zhu, X., W. Su, L. Lu, B. Li, X. Wang, and J. Dai, (2021). *Deformable DETR: Deformable Transformers for End-to-End Object Detection*. URL: `http://arxiv.org/abs/2010.04159` (visited on 01/09/2023). preprint.

TABLES AND FIGURES

Table 1: 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 accuracy's shown. EVOCNN was trained only on Fashion MNIST with the % error shown. VGG16 was trained on both datasets. Results were taken from Sun et al. (2020) and *Torchvision Main Documentation* (2023). *Number of parameters for VGG16 was miss-reported by Sun et al. (2020) as 26 million.

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

Table 2: Pros and cons of popular model architectures for image classification, object detection and semantic segmentation.

(a) Image classification architectures. Ranges of values represent the smallest and largest off-the-shelf versions available. Values for number of trainable parameters (displayed in millions (M)) and giga floating point operations (GFLOPS) were obtained from the pytorch documentation (*Torchvision Main Documentation* 2023)

Image classification			
Arch.	**n param.**	**GFLOPS**	**Pros & Cons**
ResNet (2015)	12M-60M	1.8-11.5	**Pros** ResNet18 is the smallest and most computationally efficient model here ResNet18 is ideal for modestly sized datasets ResNet152 performs comparably with transformers like VIT Widely used and tested **Cons** Uses batch normalisation which can introduce instability and inconsistent results
EfficientNet (V2) (2019)	22M-119M	8.4-56.1	**Pros** Allows the depth, width and resolution of the model to be scaled with a single coefficient **Cons** Scaling requires editing the source code Evaluation using GradCam showed much overfitting, despite high test scores
ConvNeXT (2022)	29M-198M	4.5-34.3	**Pros** Reported to outperform any architecture here and requires much less data than VIT Is scaled easily by editing the convolutional block settings Incorporates several modern features like GELU, stochastic depth and layer norm **Cons** The smallest off-the-shelf configurations are too large for many projects and may overfit Potential compatibility issues with conversion to ONNX format
VIT (2021)	87M-634M	17.6-1,016	**Pros** If trained on millions of images, VIT may slightly outperform ResNet152 **Cons** Requires huge datasets to outperform CNNs Computationally expensive to train and run at inference

Table 2: Pros and cons of popular model architectures for image classification, object detection and semantic segmentation. (continued)

(b) Object detection and semantic segmentation architectures. Ranges of values represent the smallest and largest off-the-shelf versions available. Values Faster-RCNN and Mask-RCNN were obtained from the pytorch documentation (*Torchvision Main Documentation* 2023), but values for number of trainable parameters (displayed in millions (M)) and giga floating point operations (GFLOPS) for YOLO and DETR were calculated for this comparison with an image size of 224x224 pixels.

Object detection & Semantic segmentation			
Arch.	**n param.**	**GFLOPS**	**Pros & Cons**
Faster-RCNN (2015)	44M	280.4	**Pros** Generally gives higher 'Mean Average Precision' than YOLO Performs better than YOLO on small objects Does poorly when objects overlap **Cons** More computationally expensive than YOLO
Mask-RCNN (2017)	46M	333.6	
YOLO (2016)	7M	1.01	**Pros** Extremely fast at inference time Fast to train Very easy to implement **Cons** Performs poorly on small objects Gives least accurate results of the three architectures listed here
DETR (2021)	40M	11.2	**Pros** Negates the need for region proposal and non-max suppression Performs better than Faster-RCNN and YOLO for overlapping objects As opposed to classification, transformers like DETR show promise in object detection Faster at inference than Faster-RCNN **Cons** Very computationally expensive to train Slow to converge in training Requires huge amount of training data Can be challenging to implement Requires a large batch size to achieve stable training

Table 3: Results of an ablation study to assess the effects of image normalisation and batch normalisation on a model's ability to detect plant disease

Image Norm	**Batch Norm**	**Layer Norm**	**Train time (m)**	**Loss (%)**	**Acc (%)**	**Recall (%)**	**Precision (%)**	**F1 (%)**
ConvNeXt Tiny								
No	-	Yes	1,344	0.290	88.25	88.25	88.82	88.14
Yes	-	Yes	1,368	0.322	84.51	87.51	88.14	87.38
ResNet18								
No	Yes	-	739	0.361	85.41	85.41	86.17	85.3
Yes	Yes	-	1,088	0.380	85.14	85.18	85.68	84.96
No	No	-	1,764	0.412	83.49	83.49	84.05	83.38

Table 4: Specifications and use cases for the hyperspectral cameras used in the following studies (Okamoto et al., 2007; Feng et al., 2018; Gutiérrez et al., 2019; Pan et al., 2019; Li et al., 2020; Nguyen et al., 2020).

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 w/ pushbroom	Potato yield estimation	400-1000	272	6
ImSpector N17E	Maize kernel vigour assessment	874-1734	NA	5
ImSpectorV10	Weed identification	400-1000	240	10
OCI-UAV-1000 w/ pushbroom	Nutrient assessment in rice	460-983	116	5
ImSpector V10E	Disease monitoring in pears	328-1115	1002	2.8

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

Figure 1: Application of semantic segmentation with Mask-RCNN 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.

(a) Original (b) Normalised

Figure 2: (a) Original and (b) normalised images of a cocoa pods showing various stages of disease development. Note the affect of normalisation on ones 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)

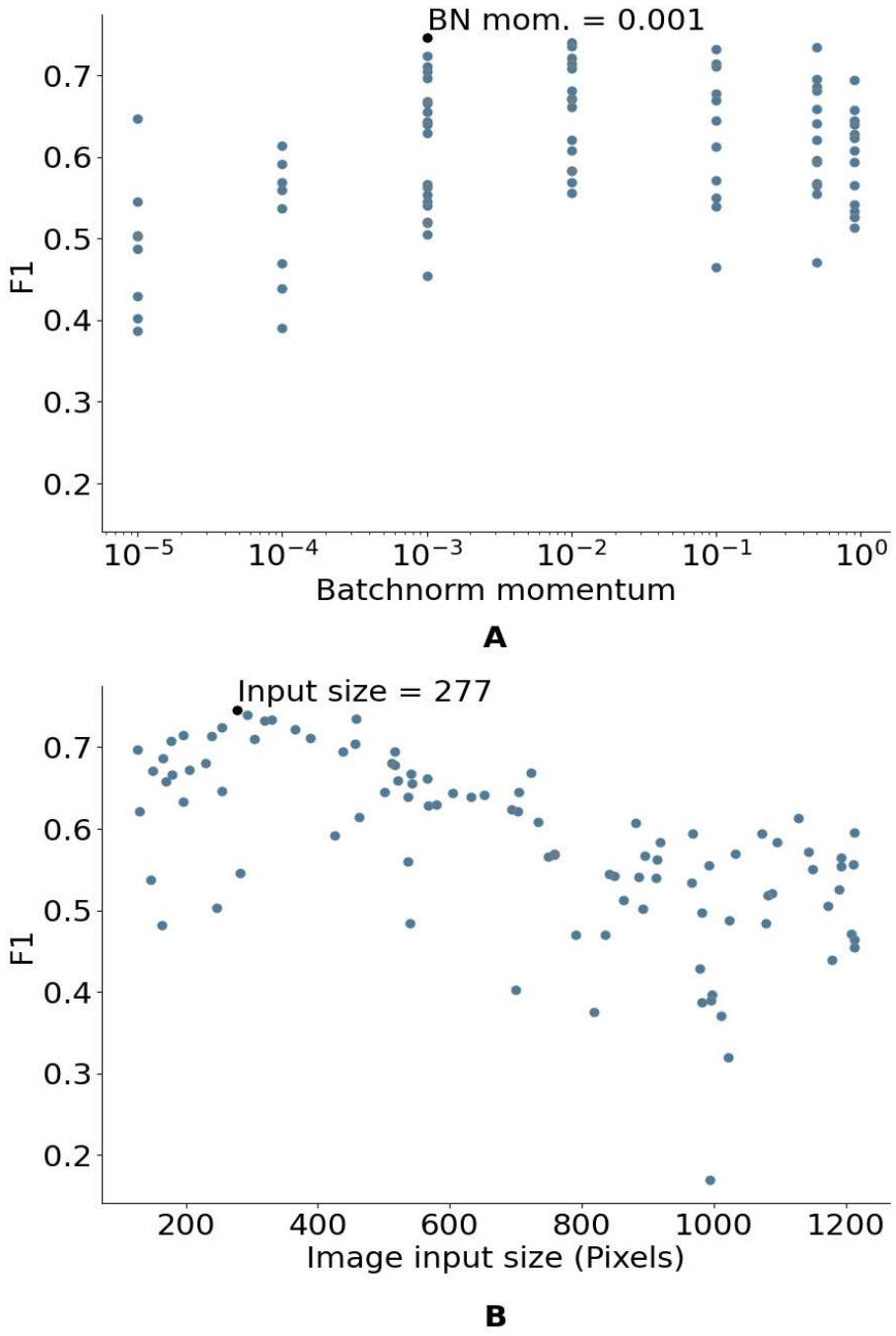

Figure 3: Results of a hyperparameter optimisation sweep training 100 ResNet18 models for disease detection in cocoa trees with variable Batchnorm momentum (A) and square image input size (B). The optimisation sweep randomly sampled from distributions of the two variables concurrently. Beginning with the ImageNet1KV2 weights, the models were trained on a dataset of 1065 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.