A new large dataset and a transfer learning methodology for plant phenotyping in Vertical Farms

Nico Sama'⋆, Etienne David◦, Simone Rossetti◇,⋆ Alessandro Antona◦, Benjamin Franchetti◦, Fiora Pirri⋆,◇
◦ Agricola Moderna, ⋆ DeepPlants, ◇ University of Rome Sapienza

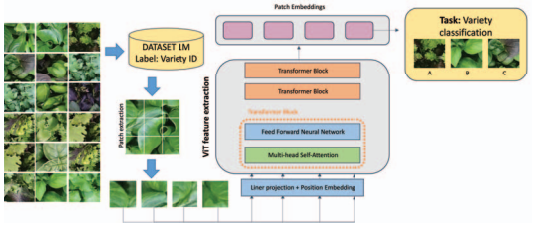
Abstract

Vertical farming has emerged as a solution to enhance crop cultivation efficiency and overcome limitations in conventional farming methods. Yet, abiotic stresses significantly impact crop quality and increase the risk of food loss. The integration of advanced automation, sensor technology, and deep learning models offers a promising solution for quality monitoring addressing the limitations of stress-specific approaches. Due to the large range of possible quality issues, there is a need for a general method. This study proposes a new plant canopy dataset, dubbed AGM of 1M images, annotated with 18 classes, an in-depth analysis of its quality for its use in transfer learning, and a methodology for detecting canopy stresses in vertical farming. The present study trains ViTbase$_8$, ViTsmall$_8$, and ResNet50 both on ImageNet and the proposed dataset on crop classification. Features from AGM and ImageNet are used for a downstream task on healthy and stress detection using a small annotated validation dataset obtaining 0.97%, 0.93%, and 0.92% best accuracy with the AGM features. We compare with standard datasets like Cassava, PlantDoc, and RicePlant obtaining significant accuracy[1]*. This research contributes to improved crop quality, prolonged shelf life, and optimized nutrient content in vertical farming, enhancing our understanding of abiotic stress management.*

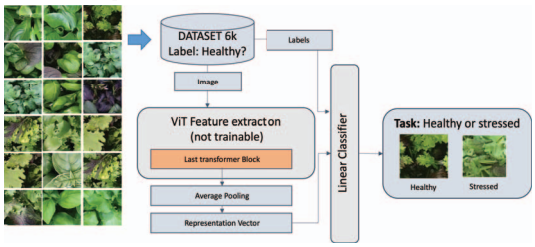
1. Introduction

Vertical farms, also known as plant factories, have emerged as a solution to improve crop cultivation response whilst maximising their nutrient content. Providing a highly protected environment for crop cultivation vertical farms enable (1) a significantly higher productivity [77], (2) a reduction in biomass waste, land usage, and water consumption, (3) growing fresh, nutritious, pesticide-free plants, regardless of location and 365 days a year. However, as all input conditions are controlled, critical parameters such as irrigation, substrate quality, pH, light intensity and quality [63, 50, 51], along with climate conditions such as airflow [34], temperature, and humidity actively induce abiotic stresses on the canopy significantly impacting crop quality, yield and exacerbating the risk of biomass loss. Detection of canopy stresses, such as excessive plant humidity, leaf nutrition deficiency, or leaf damage, throughout the growth cycle, coupled with meticulous analysis of crop management data, is crucial to mitigate the root causes of abiotic stresses. By proficiently identifying and managing canopy stresses, farmers can optimise yield, crop quality, prolong shelf life, and ensure an optimal nutrient content profile.

With the integration of advanced automation [45] and sensor technology [24], vertical farms provide active monitoring capabilities for quality assessment. Leveraging computer vision technology [24, 27, 28], and high-resolution RGB imagery, accurate and scalable monitoring in vertical farming becomes achievable through the application of computer vision [60, 72] and deep learning techniques



(a) Step 1: A network is trained to learn useful features



(b) Step 2: Features are re-used to train data efficient classifiers

Figure 1: Presentation of the main methodology

[1]The datasets will be available at https://huggingface.co/datasets/deep-plants/AGM_HS and https://huggingface.co/datasets/deep-plants/AGM; the code at https://github.com/deepplants/AGM_plant_phenotyping.

[55, 9, 7, 33]. However, the existing approaches are often confined to detecting specific stresses, limiting the ability to identify and address unknown stresses that may emerge as outliers. A more comprehensive algorithm is needed to extract meaningful features from captured images and automatically check whether a canopy is healthy or unhealthy, encompassing known and unknown stresses. Methods such as self-supervised, weakly supervised, and transfer learning [30, 14, 12] not relying on specific labelling but on very large-scale datasets are better indicated to lifting good features for stress and disease detection. Yet, to leverage these methods to plant phenotyping, adequate datasets are needed [81] to obtain separable feature spaces. In this work, we introduce a methodology showing crucial transfer learning aspects that apply well to vertical farms, but in fact, it is general. Our main contributions are summarised in Figure 1, and in the following statements:

- We present a large dataset on plant canopies exceeding currently available public plant datasets in size. See footnote 1.
- We show that the features we can extract with transformers such as ViTsmall and ViTbase, and CNNs like Resnet 50 as encoders on the crops (species and mix of them) classification task, requiring no labelling effort, have good transfer learning properties.
- We show that generating a small dataset for detecting health and stress and attaching different heads to the encoders obtains the best accuracy of 97%.
- We compare with other compelling datasets for plant health detection pretrained on ImageNet such as PlantDoc [69], Cassava [48], and RicePlant [39]. On PlantDoc and Rice plants, our method is state-of-the-art with resp. 79% and 89% accuracy on the val set, proving the competitiveness of the features obtained with our dataset.

2. Related work

Plant Stress detection Stress detection and classification are common in image-based phenotyping, [26] trained an explainable network that presented a pathologist-level performance in 2018. Approaches exploiting publicly available datasets, such as Plant Village [47], Cassava [48], Rice Leaf Disease Dataset [56], and PlantDoc [68], have achieved very high accuracy for classification. All these approaches use backbones pretrained on ImageNet. These datasets often show single leaves with a high-stress level compared to healthy ones and require backbone pretraining, being too small to be used for feature representation learning. [29] achieved the best results on three datasets: 99.39% on PlantVillage, 99.66% on Rice Leaf Disease Dataset, and 76.59% on Cassava, with a Xception-like architecture [15]. Comparable high results were obtained by [8] on Plant Village, reaching 99% accuracy, using other backbones, including Vision Transformer (ViT) [19]. They achieved 100% mean accuracy on the Wheat Rust Classification dataset and 92% on the Rice Leaf Disease dataset. PlantDoc is a more challenging benchmark, with the best result being 65.74% obtained by [61]. Many other works used deep learning to detect diseases in citrus [73], rust [62], tea leaf blight [4, 65], Northern Leaf Blight in maize plants [17], and other rice plant diseases [67]. Reviews on computer vision and machine learning methods for disease detection have been conducted in [6], [1], and [43].

Plant stress detection and segmentation This have been explored in the works of [75] and [86]. Typically, disease segmentation uses image processing techniques such as filtering, thresholding, Gaussian mixtures, and colour transforms. [5] noted that when disease symptoms exhibit colour variations compared to the surrounding areas, region of interest (ROI) segmentation can be effectively used to improve classification. This observation has led to further studies on improving disease classification through segmentation, as demonstrated by [27] and [64]. Segmenting the leaf also enhances disease classification. [2, 3] proposed automatic extended region of interest (EROI) generation through leaf segmentation, which improves detection. Another approach, as applied in the case of [83], involves dividing a large image into smaller patches for lesion classification using a sliding window technique.

Plant Stresses Specific to vertical farm In the context of plant stress detection, [66] conducted tip-burn identification in Plant Factories using GoogLeNet. They performed binary classification on single lettuce images, manually collecting images of individual plants to detect tip-burn. In [27, 24], both detection and segmentation for tip-burn on large dense canopies of indoor-grown plants were modelled. A study by [20] focused on segmentation at the canopy level for apple scab detection. The authors augmented the segmentation training set using conditional GANs to improve segmentation accuracy.

Transfer learning and self-supervised detection in plant phenotyping. Transfer learning is a well-established technique in Deep Learning, where the weights of a model trained on a specific task A are reused as a starting point for a new task B. This approach is particularly effective when task A involves training on a large dataset, with popular choices being ImageNet [37]. Several studies [47, 76, 71, 23] demonstrated that transfer learning from ImageNet leads to superior performance compared to training models from scratch. In a study by [41], an alternative large dataset called PlantCLEF 2015, with more than 2 million images but only 100 images per species, is considered. PlantCLEF 2015 was used to pretrain a network for disease detection. The authors showed that a plant-specific dataset reduces the risk of overfitting compared to ImageNet. Self-supervision, another approach in transfer learning, has been investigated by [49], where they

Figure 2: The figure shows the 18 crop types from the AGM$_{SP}$ dataset with their index name.

have tested self-supervised methods such as SimCLR[14], SwAV[12] and Barlow Twins [84] to check the efficacy of self-supervision on the soybean plant stress dataset; however, they applied end-to-end fine-tuning evaluation. Self-supervised methods have dealt with ImageNet features so far, and it is hard to believe that ImageNet can provide optimal features for plant phenotyping downstream tasks. A study by [52] compared self-supervised learning methods with supervised alternatives for plant phenotyping downstream tasks and found that the self-supervised methods performed worse. The study suggests that dedicated feature extraction methods and datasets for plant phenotyping are still relevant problems for self-supervised deep learning.

3. Motivations

Deep-learning methods for plant phenotyping need access to well-defined features to meaningfully lift segmentation and classification methods to face a large number of challenging missions on plants, which often can exploit just a few thousand images. And despite as in any other task, pretraining on ImageNet [18] is better than training from scratch [47], this seems to be verified only on datasets with clear separation of disease features such as PlantVillage [32] and Cassava [48]. However, it cannot be generalized to any plant phenotyping task, considering Imagenet is conceived for everyday human tasks, as shown in [41, 46].

Yet, gaining the extraordinary vastity of classes and well-curated features such as ImageNet is very hard, and few attempts have been made so far [13]. In this work, we introduce a new giant dataset of plant canopies collected in a Vertical Farm with a limited number of classes, namely the plant species, and discuss basic pretraining methods and the features space of the dataset, comparing the emerging properties to the ImageNet ones. Furthermore, we show that features extracted from the proposed dataset can be used on a small dataset for detecting healthy and stressed plants obtaining accuracy scores better than ImageNet, on the same dataset. Comparable results on well-known public datasets for disease detection, such as PlantDoc [69], Cassava [48], and RicePlant [39] show that despite our dataset being limited to vertical farms canopies, it is already comparable to ImageNet for plants stress and disease detection.

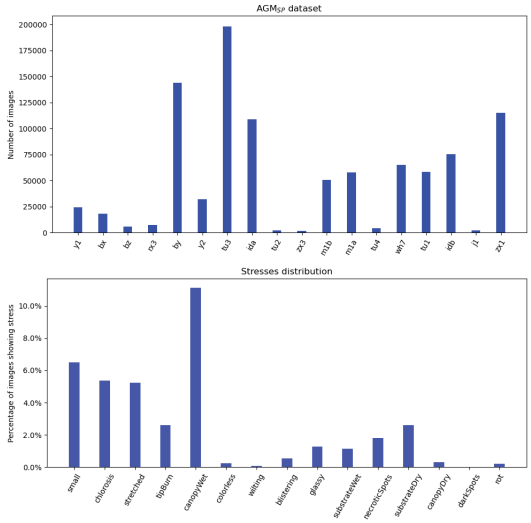


Figure 3: Above: Class frequency distribution of AGM. Below: Prevalence of stresses on AGM dataset.

4. Dataset

We introduce a dataset, named AGM, meaning AGricultureModern, consisting of 972,858 120×120 RGB images, encompassing 18 plant crops, here a crop is a species or a mix of species. Figure 3 (above), provides an overview of how the images are distributed among the different crops, where scarcely represented crops are used as tests. Within certain classes, there are variations in crops or different varieties present. For example, crops *bz, by*, and *bx* represent different varieties of basil, while crops *tu1, j1, zx3* are different varieties of Batavia lettuce. Additionally, some crops feature a combination of two or three different species mixed together, e.g. *m1a, m1b, idb, ida* and *m1a* and *idb* share the same mixture but in different proportions, as well as *m1b* and *ida*. Further insights into these mixes can be found in Figure 2. During harvesting, trays of different sizes are assembled in a grid and placed over a moving table. The images of the full table were captured using a high-

resolution camera over a period spanning from May 2022 to December 2022. It is worth noting that the illumination projects light patterns on the plants, resulting in a high variance of shade and light. During harvesting, agronomists utilized specialized software to divide the table images into images of size of 1073×650 pixels of the original trays, with $1px$ corresponding to $0.5mm$, and annotated the crops as well as the type and severity of any observed stress, assigning it a numerical score (0 to 3). A breakdown of the most common types of stresses found in the dataset is shown in Figure 3.

Preprocessing and Pretraining We set up a supervised classification task for crops classification on AGM and train a model composed of a large encoder E and a classification head H by optimizing a cross-entropy loss between the model predictions and the ground truth labels. For each mini-batch $(\mathbf{x}, \mathbf{y})$, the learning objective is:

$$\mathcal{L} = -\sum_i y_i \log(H(E(x_i))) \tag{1}$$

Successively, we discard the head H and evaluate the effectiveness of pretraining with our dataset by treating the encoder E as a feature extractor and analyse the quality of the extracted features by adapting them to new plant-disease domains. The architecture of the model encoder E is either based on a deep convolutional residual neural network (ResNet50 [31]) or transformer-based architectures (ViT base_8 and ViT small_8 [19]), which have shown promising performance in computer vision tasks. During pretraining, we employed a simple classification head consisting of a 3-layers MLP with GELU non-linearities and hidden dimension halved at each layer before the last one. We utilized the Adam [35] optimizer with a learning rate of 10^{-3} and train for 100 epochs.

Table 1 presents the training, validation, and test accuracies for crops classification using three different encoders and the MLP classification head. The encoders are pretrained on either ImageNet or AGM. We observe that the ViT models consistently outperform the ResNet50 architecture in terms of both training and validation accuracy. Interestingly, the accuracy achieved by the ViT models does not exhibit a strong dependence on the model size. Both ViT base_8 and ViT small_8 attain comparable training and validation accuracy, indicating that the smaller variant can effectively learn representations despite its reduced capacity. Moreover, despite employing a simple classification head and encoder architectures, we achieve results that are better than encoders pretrained on ImageNet, which could point to the effectiveness of AGM pretrained models in capturing the unique characteristics of plant images.

To test if the generalization capabilities of the model are enhanced by augmentation strategies with varying degrees of strength, different configurations were applied during training. These strategies included:

Encoder	Pre-trained on	Training Acc	Validation Acc.	Test Acc.
ViT small_8	AGM*	0.9965	0.9720	0.9784
ViT small_8	ImageNet	0.9305	0.9309	0.9298
ViT base_8	AGM*	0.9950	0.9702	0.9798
ViT base_8	ImageNet	0.9330	0.9303	0.9312
ResNet50	AGM*	0.9610	0.9440	0.9410
ResNet50	ImageNet	0.9512	0.9145	0.9243

Table 1: Training, validation and test accuracies for crops classification using three different encoders (ViT small8, ViT base8, and ResNet50) with a MLP classification head. The encoders are pretrained on either ImageNet or AGM. Here $*$ indicates the best in class.

■ **Rotations and Flippings**: Random rotations and horizontal/vertical flips were applied to introduce geometric variations.

■ **Mixup** [85]: Mixup augmentation combines pairs of training samples to generate interpolated images; specifically, for a pair of input images and labels (x_i, y_i), (x_j, y_j) Mixup generates new inputs $(\hat{x}, \hat{y})$, such that

$$\begin{aligned} \hat{x} &= \mu x_i + (1-\mu)x_j \\ \hat{y} &= \mu y_i + (1-\mu)y_j \\ &\text{with hyperparameter } \mu \in [0,1]. \end{aligned} \tag{2}$$

■ **Randaugment[16] no Solarization and Posterization**: We used Randaugment technique to apply a sequence of random image transformations, excluding solarization and quantization.

Model	Augmentation	Training Acc.	Val Acc.
ViT base_8	Randaug, Mixup	0.995	0.970
ViT small_8	Randaug, Mixup	0.970	0.962
ViT small_8	Rotations & Flippings	0.995	0.972
ViT small_8	None	0.996	0.968
ResNet50	Randaug, Mixup	0.951	0.914

Table 2: Pretraining accuracy for *train* and *val* sets of AGM

The selected augmentation strategies address the challenges of plant images. Stronger augmentations could hinder learning faithful representations from a very specific dataset with high self-similarity. Such datasets have a clustered data distribution, and applying strong augmentations may generate samples too far from the actual distribution, resulting in poorer generalization. Results in Table 2 confirm this hypothesis. Excessively strong augmentations, like Randaugment and Mixup, lead to a slight decrease in validation accuracy for the ViT small_8 model compared to milder augmentations. These findings emphasize the need

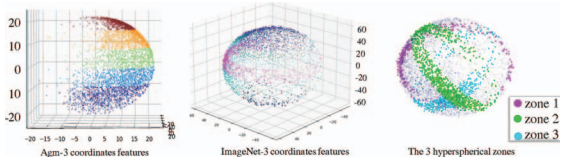


Figure 4: Coordinates of features projected on S_2, for visualization purpose. The last sphere shows the 3 zones chosen to compute the distributions on S_2.

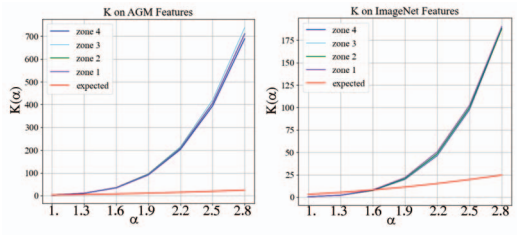


Figure 5: $K(\alpha)$ with α varying in (1,3), we can observe that by a smaller increment of arc length between points, the number of features points clustering together augment exponentially for the AGM features, while the growth for ImageNet is much slower.

to strike a balance between variations and preservation of the inherent characteristics of the original images [22].

5. Features representation analysis

Features representation has been shown to be pivotal for semi-supervised, self-supervised, and transfer learning [30, 14, 12, 46, 11]. Most weakly or self-supervised methods rely on feature metrics based on the cosine-similarity between two vectors of features $F_i, F_j \in \mathbb{R}^N$ [53]. For example, with ViT the feature vectors could be the class token cls, with $cls \in \mathbb{R}^{384}$, with small$_8$, and $cls \in \mathbb{R}^{768}$ with ViT base$_8$. Cosine similarity compares pairs of feature vectors on a unit hypersphere S_{N-1} [79], image by image, but it cannot compare sets of image features to study specific properties of the induced feature space.

The purpose of this section is to show relevant differences between ImageNet and AGM features representation on features extracted by ViT as shown in Section 4. These effects have advantages or drawbacks, according to the transfer learning method used: linear (e.g. linear classifier or K-means) versus nonlinear (e.g spectral clustering, kernel PCA), and the task at hand.

Clearly, a vector of size N lies in a Euclidean space of size N, and each specific feature is a coordinate in this space. Since N can be very large, to study the feature space induced by a dataset with M images we need to consider a group of $k < N$ features coordinates of M and map them on a manifold to analyse the distribution of sets of image features with respect to the considered coordinates. Here we choose the sphere S_k with $k < N$, namely the hypersphere in the Euclidean space $k+1$, as the manifold of interest.

To better illustrate the idea consider the image shown in Figure 4. Here we consider three random feature coordinates (for visualization purposes) from the feature average pooling of ViT base$_8$ trained on AGM and on ImageNet, and project them on the ordinal sphere S_2 in $\mathbb{R}^3$. We can note that while AGM features (the left sphere) are concentrated in an area, they lack points on a big part of the sphere. On the other hand, ImageNet (the central sphere) features are quite sparse but cover the whole sphere. Here, all visible points are the projection of a randomly selected triple of coordinates of the features of the whole dataset, and each point $\mathbf{p_j}$ is the projection of three sampled coordinate features of the j-th image.

Projection on the hypersphere Given the set of M images from the dataset and the corresponding features set $X \in \mathbb{R}^{M \times N}$, let $k \in \{3, \dots, n\}, n << N$ define a group of feature coordinates. Let $\mathbf{q}_j = (x_{i_0}, \dots, x_{i_k})$ be the chosen set of coordinates of the feature vector $\mathbf{v} \in \mathbb{R}^N$ of an image j in M, with i indicating the randomly chosen tuple of coordinates. Consider a hypersphere in $\mathbb{R}^k$ with center $\mathbf{w} \in \mathbb{R}^k$ with $w = \mathbf{0}_k$, and ray r, defined according to the set of features X, namely, $r = \frac{1}{2}(min\{X\} + max\{X\})$. The point $\mathbf{p}_j$ on the hypersphere surface, corresponding to the j-th image feature vector X_j, is defined:

$$\mathbf{p}_j = \left(\frac{r}{\|\mathbf{q}_j\|} \right) \mathbf{q}_j \tag{3}$$

Note that because $\mathbf{w} = \mathbf{0}_k$ no translation is used in the mapping. Given k, the surface area of the hypersphere is:

$$S_{k-1}(r) = \frac{2\pi^{k/2}}{\Gamma(\frac{k}{2})} r^{k-1}, \quad \Gamma \text{ is the gamma-function} \tag{4}$$

Distribution of tuples of features on the hypersphere Given k-tuples of coordinates of features randomly sampled from a set of 600 k-tuple both for the ImageNet and AGM features, we want first to assess how close they are to the uniform distribution.

Using the Kullback-Leibler divergence [38], namely $D_{KL}(f_s \| g)$, $s \in \{ImageNet, AGM\}$ with f_s a kernel density on the $k-1$ hypersphere, we evaluate how close the two distributions are from the uniform distribution $g \sim U(0,1)$, see Table 3. To make this meaningful in terms of spatial distribution, we consider k-zones. These zones (see the sphere on the right in Figure 4, showing three zones) are defined on k-great circles of the $k-1$-sphere and have a bandwidth of dimension α. We defined the spherical zones on the hypersphere as follows. Let B_I be the regularized incomplete beta-function, an hyperspherical cap A_h with height h is:

$$\begin{aligned} \text{Let:} \quad & h = r - \alpha/2 \text{ and } \nu = \tfrac{(2rh - h^2)}{r^2} \quad \textit{then} \\ & A_h = \tfrac{1}{2} S_k B_I \left(\nu; \tfrac{k-1}{2}, \tfrac{1}{2} \right) \end{aligned} \tag{5}$$

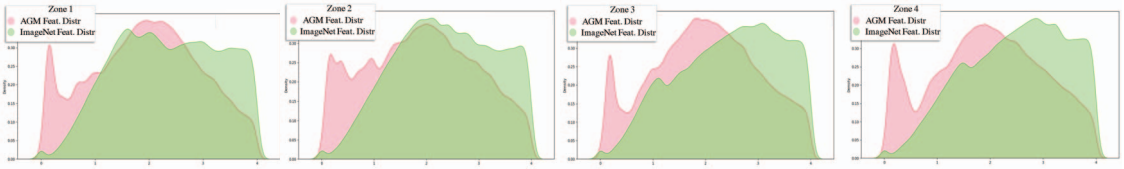


Figure 6: Kernel density estimation of points on the $K = 4$ zones of the k-coordinate features for both ImageNet and AGM features. Features were extracted by average pooling the last block of ViTbase$_8$, and projected on the $k-1$-sphere. We note the different shapes: while KDE for ImageNet features is smooth and in zone 1 close to a uniform distribution, the KDE for AGM features looks more like a mixture.

A k-spherical zone results from the difference between the hypersphere surface and two equal spherical caps:

$$S_{zone} = S_k - 2A_h \tag{6}$$

For the kernel density estimation (KDE) [25], we used the Gaussian kernel over the points within the k S_{zone}s. The density plots shown in Figure 6 help to visually check that ImageNet features have a distribution very close to the uniform, differently from the AGM ones. See also Table 3.

We further consider the $K(\alpha)$ function [59], adapted to the hypersphere. Here α is the 'width' of a zone (see Figure 4), hence of the hyperspherical cap ray, tangent to the zone. The $K(\alpha)$ function describes spatial events at varying distances, it differs from KNN since it provides a synthetic analysis of multi-distance spatial clustering. Here we use an approximate adaptation to the n-sphere, considering the hyperspherical caps with radius α on each zone surface. Since a cover of the caps with radius α on a zone cannot be complete, the resulting K is approximate. However, it is quite useful (see Figure 5) to see whether at increasing distances there is a dispersion of the points or not: given the expected value which should be $\pi\alpha^2$. If the observed values are above the graph of the expected one, the points cluster (namely more and more points belong to the same region) and are dispersed if the observed values are below. Let S_{zone} the surface area and A_h the surface of hypercaps with radius α, an approximation of K, with I the indicator function, and $\ell = r\cos^{-1}(\mathbf{p}_i\mathbf{p}_j^\top/R^2)$ the arc length, is:

$$K(\alpha) = \frac{A_{zone}}{S_k}\sum_{i\neq j} I(\ell(\mathbf{p}_i, \mathbf{p}_j) < \alpha) \tag{7}$$

Given the nature of plant images, we have seen via the $K(\alpha)$ function (see also Figure 5) that a small increase in arc length (or decrease in similarity) accumulates a huge number of features in a small region of the feature space, therefore usual separation and augmentations methods (see also Section 4) might not be able to separate the features. On the other hand, AGM features generate a distribution quite well capturing the idiosyncratic properties of the plants (see Figure 6) with respect to the task at hand, as opposed to ImageNet features generating a distribution close to the Uniform, see also Table 3. This is mirrored in our results on transfer learning for classifying healthy and stressed canopies, see the next sections.

Table 3: ImageNet and AGM feature sets

Dataset	Zone	# coord	divergence from $g \sim U(0,1)$	K-test
AGM	Zone 1	4	2.5592	2.4701
Imagenet	Zone 1	4	0.1636	0.3804
AGM	Zone 2	4	2.6455	2.5498
Imagenet	Zone 2	4	0.1530	0.4436
AGM	Zone 3	4	2.7846	2.7118
Imagenet	Zone 3	4	0.1737	0.4275
AGM	Zone 4	4	2.6113	3.7714
Imagenet	Zone 4	4	0.1619	0.4360

The table shows comparisons of ImageNet and AGM features projected on a k-sphere, as sampled from two types of distributions along k-zones of the sphere (column 2). In column 2, the number of coordinate features used is reported; in column 3, the divergence of the distribution from the uniform distribution, in column 4 the distribution of points at arc length 1, in terms of features, about the considered zones (see Figure 5). Tests are done on 600 random 4-tuple for all points on each zone.

6. Transfer Learning and Classification

A Healthy Stressed validation set In some cases, related to our dataset AGM, instances of stress may not be easily visible in the top-view images. Additionally, each tray contains multiple plants and has a size of 1073×650 pixels, making it challenging to accurately identify small areas of stress within the full image or detect multiple stresses in a single image. To overcome these challenges, a secondary stage of annotation is conducted. During this stage, labelers extract 120×120 sub-images from the tray images selecting those ones showing clear signs of either good health or high stress, resulting in the creation of a smaller subset of the original dataset comprising 6,127 images. This subset consists of 3,798 healthy samples and 2,329 stressed samples across 14 of the 18 classes of AGM. For this small subset, which we name AGM$_{HS}$, labelers collected images by clicking on a point on the stressed leaves. The collected clicked points were used as prompts to Segment Anything [36] to semi-automatically generate masks of the stressed areas. See Figure 7, showing examples of the extracted masks and relative stress. Together with the healthy/stressed classification labels for AGM$_{HS}$ we also release the segmentation masks. This addition to the dataset enables the development and evaluation of advanced segmentation

Figure 7: Samples from AGM_{HS} dataset and associated masks highlighting tip-burn and chlorosis.

Encoder	Pre-trained on	Val. Acc. (MLP)	Val. Acc. (Linear)	Val. Acc. (KNN)
ViT $small_8$	AGM	**0.9720**	0.8300	**0.9502**
ViT $small_8$	ImageNet	0.9478	**0.9461**	0.8702
ViT $base_8$	AGM	**0.9273**	**0.9404**	**0.9102**
ViT $base_8$	ImageNet	0.9078	0.8321	0.8989
ResNset50	AGM	**0.9142**	**0.8800**	**0.8637**
ResNset50	ImageNet	0.8625	0.8446	0.8230

Table 4: ViT $small_8$, ViT $base_8$ and Resnet50 accuracies on the healthy and stress task using 3 different heads, namely MLP, Linear, and KNN fine-tuned on the small dataset of 6000 HS annotated images.

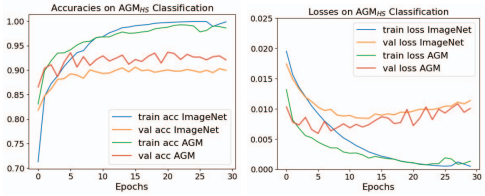


Figure 8: Comparison of training and validation accuracies (left) and losses (right) for AGM_{HS} and ImageNet classification

models specifically designed for detecting and localizing plant stress in top-view images. Indeed, the top-view perspective of these images presents an interesting challenge and opportunity for segmentation models. In most cases, the majority of the image area is covered by the top view of healthy leaves, effectively serving as the background information for a segmentation model. While there are additional elements present in the images, such as the terrain or pieces of the table structure, these represent a small but non-negligible area in comparison.

In future research we shall show results using the segmentation components of the dataset, in this work, we do not actually use them, focusing on the transfer learning from the AGM extracted feature set, for classification.

Evaluation of learned representations In our evaluation of the learned representations [21] from AGM pretraining models compared to their ImageNet counterpart, we focus on the task of healthy-stressed classification. As discussed in Section 5 here we show that AGM-pretrained models exhibit better discriminative capabilities and are more adept at capturing the nuances of stressed regions compared to the ImageNet-pretrained models [78].

Our approach involves attaching simple classification heads, including an MLP as described in Section 4, a linear classifier, and KNN, to the pretrained encoder E. To train the linear and MLP heads, we use cross-entropy loss and employed the Adam optimizer with a learning rate of 10^{-4}. For the Vision Transformer encoders (ViTs), the MLP-heads utilize the *cls* token, the linear classifier uses the average pooling of the *cls* token with the last features block, while for the ResNet50 encoder, we use the flattened output from the last layer. The results presented in Table 4 demonstrate that AGM pretraining consistently outperforms ImageNet pretraining across the various classification heads; in particular, among the tested classifiers, the 3-layer MLP achieved the best performance. For the ViT $small_8$-MLP configuration, we also show the dynamics of training and validation losses and accuracies throughout the training process and their comparison with ImageNet in Figure 8.

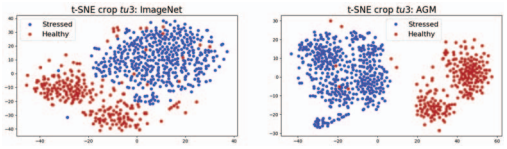


Figure 9: Comparison of t-SNE 2-D projection of features from ImageNet and AGM pretrained models for crop *zx1* samples (above) and crop *zx*1 samples (below), from AGM_{HS}.

This finding suggests that leveraging a dataset with domain-specific characteristics and features, such as AGM, can lead to improved performance and better alignment with the target task [10]. In the context of the healthy-stressed classification task, our findings show the benefits of pretraining on AGM for detecting variations and accurately identifying stressed areas within plant samples. We conduct an analysis of the interclass separation between healthy and stressed plant samples on the AGM_{HS} dataset to elucidate the role of the pretraining source. Separability of the frozen features from the pretrained ViTs via clustering and dimensionality reduction techniques with respect to our AGM dataset and ImageNet is shown in Figure 9.

Implementation details The experiments were conducted on a computing setup consisting of two NVIDIA A6000 GPU and utilizing the PyTorch [54] deep learning framework. We trained with a batch size of 64, where the models were pretrained for 100 epochs and fine-tuned for 30 epochs. The dataset used for pretraining consisted of nearly one million RGB images. For evaluation, three distinct datasets were utilized: PlantDoc [69], Cassava [48], and RicePlant [39] datasets. During the training process, a validation split of 20% was employed for all experiments. Addi-

Dataset	Model	Multi-class Val Acc.
Cassava	T-RNet [87]	0.9112
Cassava	ECNN [42]	0.8870
Cassava	A.M. EfficientNet [58]	0.8708
Cassava	FormerLeaf [74]	**0.9500**
Cassava	DenseNet121 [70]	0.8786
Cassava	Ours (ViT small$_8$)	0.9377
PlantDoc	InceptionResNet V2 [69]	0.7053
PlantDoc	DenseNet201 [57]	0.6718
PlantDoc	Ours (ViT small$_8$)	**0.7972**
RicePlant	VGG16 [40]	0.7312
RicePlant	Ours (ViT small$_8$)	**0.8905**

Table 5: Accuracy for a fine-tuned ViTsmall$_8$ on Cassava, RicePlant and PlantDoc; comparison with state-of-the-art.

Dataset	Pre-trained on	HS Validation Accuracy
Cassava	AGM	0.8920
Cassava	ImageNet	**0.9065**
PlantDoc	AGM	**0.6674**
PlantDoc	ImageNet	0.6566
RicePlant	AGM	0.8905
RicePlant	ImageNet	**0.9154**

Table 6: Accuracy for a fine-tuned ViTsmall$_8$ on Cassava, RicePlant, PlantDoc. For each dataset we formulated a binary classification for healthy and stressed samples.

tionally, for the AGM$_{HS}$ experiments, also test split of 20% was reserved. The ResNet model used in this study was based on the Torchvision [44] implementation, while ViT was adapted from the timm [80] library. Experiments comparing AGM pretraining with ImageNet pretraining used weights provided by [82].

7. Comparisons with state of the art methods

In this section, we present the validation accuracy results obtained by fine-tuning a ViT small$_8$ model pretrained on our dataset and compare them with other state-of-the-art methods. We provide a comprehensive analysis by considering three plant-based datasets, specifically PlantDoc [69], Cassava [48], and RicePlant [39], and highlight any important considerations for fair comparisons. For more details see 1. Table 5 summarizes the validation accuracy results for the fine-tuned ViT small$_8$ model on our dataset, alongside benchmarked results from other methods. Notably, for the Cassava dataset, we compare with papers that utilize the same extended dataset version of more than 21,000 images and report results on the validation set. Additionally, we focus on comparing our results with methods applied to the *imbalanced* version of the Cassava dataset to ensure consistency and fairness in the comparison.

Regarding the PlantDoc dataset, we consider methods that train on the entire images, while some studies focus on the *cropped* version of the dataset where images are cropped along the annotation bounding boxes. The fine-tuned ViT small$_8$ model demonstrates promising results and competes well against other state-of-the-art methods. On the Cassava dataset, our approach outperforms all CNN-based methods in terms of multi-class validation accuracy, with the exception of the transformer-based method FormerLeaf [74]. Our approach also exhibits competitive performance on the PlantDoc dataset, demonstrating efficacy while maintaining simplicity and efficiency in the model architecture compared to more complex methods [88]. On the RicePlant dataset, our approach surpasses the current state-of-the-art method proposed by Kumar et al. [40]. However, it is important to note the limited availability of studies and exploration on the RicePlant dataset for comprehensive comparisons. Overall, the fine-tuned ViT small$_8$ model showcases its competitiveness and promising performance across multiple plant-based datasets, positioning it as an effective and efficient approach for agricultural computer vision tasks.

We conduct a binary healthy/stressed classification task on the three datasets considered, with the objective to classify images as either healthy or stressed, disregarding the specific diseases and plant crops present. In Table 6 we report results for this binary classification task. In comparison to an equivalent model pretrained on ImageNet, our model achieved comparable results. Specifically, on the PlantDoc dataset, our pretrained model outperformed the ImageNet-pretrained model by 1.6% in terms of classification accuracy. However, on the Cassava and RicePlant datasets, the ImageNet-pretrained model had a slight advantage, surpassing our model by 1.6% and 2.7%, respectively.

8. Conclusions

This work demonstrates the potential of pretraining vision transformers on large-scale, domain-specific datasets for agricultural computer vision tasks, focusing on vertical farming. Leveraging a novel dataset, AGM, of nearly 1 million canopy images of size 120 × 120, we reveal the superiority of features learned from the domain-specific dataset, especially in stressed vs healthy plant classification. The analysis highlights differences in feature space distributions between ImageNet and the domain-specific dataset, indicating better nuances capturing in plant images. Fine-tuning the AGM pretrained model on public datasets like PlantDoc, RicePlant, and Cassava achieves state-of-the-art accuracy for plant disease detection. Future research should optimize the model architecture, explore dataset-specific techniques, and integrate domain knowledge for enhanced plant species classification and health state detection algorithms. This approach promises effective computer vision models for plant health monitoring and most phenotyping tasks based on high-resolution RGB imagery.

References

[1] Andre S Abade, Paulo Afonso Ferreira, and Flavio de Barros Vidal. Plant diseases recognition on images using convolutional neural networks: A systematic review. *arXiv preprint arXiv:2009.04365*, 2020. 2

[2] Aliyu Muhammad Abdu, M Mokji, and UU Sheikh. An investigation into the effect of disease symptoms segmentation boundary limit on classifier performance in application of machine learning for plant disease detection. *International Journal of Agricultural, Forestry & Plantation (ISSN No: 2462-1757)*, 7(6):33–40, 2018. 2

[3] Aliyu Muhammad Abdu, Musa Mohd Mokji, and Usman Ullah Sheikh. An automatic plant disease symptom segmentation concept based on pathological analogy. In *2019 IEEE 10th Control and System Graduate Research Colloquium (ICSGRC)*, pages 94–99. IEEE, 2019. 2

[4] Wenxia Bao, Tao Fan, Gensheng Hu, Dong Liang, and Haidong Li. Detection and identification of tea leaf diseases based on ax-retinanet. *Scientific Reports*, 12(1):2183, 2022. 2

[5] Jayme Garcia Arnal Barbedo. A new automatic method for disease symptom segmentation in digital photographs of plant leaves. *European journal of plant pathology*, 147(2):349–364, 2017. 2

[6] Jayme Garcia Arnal Barbedo. Detection of nutrition deficiencies in plants using proximal images and machine learning: A review. *Computers and Electronics in Agriculture*, 162:482–492, 2019. 2

[7] Anjanadevi Bondalapati. An improved deep learning model for plant disease detection. 07 2020. 2

[8] Yasamin Borhani, Javad Khoramdel, and Esmaeil Najafi. A deep learning based approach for automated plant disease classification using vision transformer. *Scientific Reports*, 12(1):11554, 2022. 2

[9] Med Brahimi, Marko Arsenovic, Sohaib Laraba, Srdjan Sladojevic, Boukhalfa Kamel, and Abdelouahab Moussaoui. *Deep Learning for Plant Diseases: Detection and Saliency Map Visualisation*. 06 2018. 2

[10] Manh-Ha Bui, Toan Tran, Anh Tran, and Dinh Phung. Exploiting domain-specific features to enhance domain generalization. *Advances in Neural Information Processing Systems*, 34:21189–21201, 2021. 7

[11] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. pages 132–149, 2018. 5

[12] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *NeurIPS*, 33:9912–9924, 2020. 2, 3, 5

[13] Julien Champ, Titouan Lorieul, Maximilien Servajean, and Alexis Joly. A comparative study of fine-grained classification methods in the context of the lifeclef plant identification challenge 2015. In *CLEF: Conference and Labs of the Evaluation forum*, number 1391, 2015. 3

[14] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. pages 1597–1607. PMLR, 2020. 2, 3, 5

[15] François Chollet. Xception: Deep Learning with Depthwise Separable Convolutions, Apr. 2017. arXiv:1610.02357 [cs]. 2

[16] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703, 2020. 4

[17] Chad DeChant, Tyr Wiesner-Hanks, Siyuan Chen, Ethan Stewart, Jason Yosinski, Michael Gore, Rebecca Nelson, and Hod Lipson. Automated identification of northern leaf blight-infected maize plants from field imagery using deep learning. *Phytopathology*, 107:1426–1432, 06 2017. 2

[18] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 3

[19] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, June 2021. arXiv:2010.11929 [cs]. 2, 4

[20] Clément Douarre, Carlos F Crispim-Junior, Anthony Gelibert, Laure Tougne, and David Rousseau. Novel data augmentation strategies to boost supervised segmentation of plant disease. *Computers and electronics in agriculture*, 165:104967, 2019. 2

[21] Cian Eastwood and Christopher KI Williams. A framework for the quantitative evaluation of disentangled representations. In *International conference on learning representations*, 2018. 7

[22] Mohamed Elgendi, Muhammad Umer Nasir, Qunfeng Tang, David Smith, John-Paul Grenier, Catherine Batte, Bradley Spieler, William Donald Leslie, Carlo Menon, Richard Ribbon Fletcher, et al. The effectiveness of image augmentation in deep learning networks for detecting covid-19: A geometric transformation perspective. *Frontiers in Medicine*, 8:629134, 2021. 5

[23] Konstantinos P Ferentinos. Deep learning models for plant disease detection and diagnosis. *Computers and electronics in agriculture*, 145:311–318, 2018. Publisher: Elsevier. 2

[24] Benjamin Franchetti and Fiora Pirri. Detection and localization of tip-burn on large lettuce canopies. *Frontiers in Plant Science*, 13, 2022. 1, 2

[25] Keinosuke Fukunaga and Larry Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on information theory*, 21(1):32–40, 1975. 6

[26] Sambuddha Ghosal, David Blystone, Asheesh K. Singh, Baskar Ganapathysubramanian, Arti Singh, and Soumik Sarkar. An explainable deep machine vision framework for plant stress phenotyping. *Proceedings of the National Academy of Sciences*, 115(18):4613–4618, May 2018. 2

[27] Riccardo Gozzovelli, Benjamin Franchetti, Malik Bekmurat, and Fiora Pirri. Tip-burn stress detection of lettuce canopy

grown in plant factories. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1259–1268, 2021. 1, 2

[28] Munirah Hayati Hamidon and Tofael Ahamed. Detection of tip-burn stress on lettuce grown in an indoor environment using deep learning algorithms. *Sensors*, 22(19), 2022. 1

[29] Sk Mahmudul Hassan and Arnab Kumar Maji. Plant disease identification using a novel convolutional neural network. *IEEE Access*, 2022. 2

[30] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. pages 9729–9738, 2020. 2, 5

[31] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[32] David. P. Hughes and Marcel Salathe. An open access repository of images on plant health to enable the development of mobile disease diagnostics, 2016. 3

[33] Peng Jiang, Yuehan Chen, Bin Liu, Dongjian He, and Chunquan Liang. Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. *IEEE Access*, 7:59069–59080, 2019. 2

[34] Christopher Kaufmann. *Reducing Tipburn in Lettuce Grown in an Indoor Vertical Farm: Comparing the Impact of Vertically Distributed Airflow vs. Horizontally Distributed Airflow in the Growth of Lactuca sativa*. PhD thesis, The University of Arizona, 2023. 1

[35] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[36] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023. 6

[37] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017. 2

[38] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951. 5

[39] Raj Kumar. Rice plant dataset, 2020. 2, 3, 7, 8

[40] Raj Kumar, Gulsher Baloch, Pankaj, Abdul Baseer Buriro, and Junaid Bhatti. Fungal blast disease detection in rice seed using machine learning. *International Journal of Advanced Computer Science and Applications*, 12(2), 2021. 8

[41] Sue Han Lee, Hervé Goëau, Pierre Bonnet, and Alexis Joly. New perspectives on plant disease characterization based on deep learning. *Computers and Electronics in Agriculture*, 170:105220, 2020. 2, 3

[42] Umesh Kumar Lilhore, Agbotiname Lucky Imoize, Cheng-Chi Lee, Sarita Simaiya, Subhendu Kumar Pani, Nitin Goyal, Arun Kumar, and Chun-Ta Li. Enhanced convolutional neural network model for cassava leaf disease identification and classification. *Mathematics*, 10(4), 2022. 8

[43] Jinzhu Lu, Lijuan Tan, and Huanyu Jiang. Review on convolutional neural network (cnn) applied to plant leaf disease classification. *Agriculture*, 11(8):707, 2021. 2

[44] TorchVision maintainers and contributors. Torchvision: Pytorch's computer vision library. https://github.com/pytorch/vision, 2016. 8

[45] William Marchant and Sabri Tosunoglu. Robotic implementation to automate a vertical farm system. In *Proceedings of the 30th Florida Conference on Recent Advances in Robotics*, pages 11–12, 2017. 1

[46] H Minyoung, Pulkit Agrawal, and Alexei Efros. What makes imagenet good for transfer learning? *arXiv e-prints*, pages arXiv–1608, 2016. 3, 5

[47] Sharada Mohanty, David Hughes, and Marcel Salathe. Using deep learning for image-based plant disease detection. *Frontiers in plant science*, 7:1419, 2016. 2, 3

[48] Ernest Mwebaze, Timnit Gebru, Andrea Frome, Solomon Nsumba, and Jeremy Tusubira. icassava 2019fine-grained visual categorization challenge, 2019. 2, 3, 7, 8

[49] Koushik Nagasubramanian, Asheesh Singh, Arti Singh, Soumik Sarkar, and Baskar Ganapathysubramanian. Plant phenotyping with limited annotation: Doing more with less. *The Plant Phenome Journal*, 5(1):e20051, 2022. 2

[50] Thi Kim Loan Nguyen, Kye Man Cho, Hee Yul Lee, Du Yong Cho, Ga Oun Lee, Seong Nam Jang, Yongki Lee, Daesup Kim, and Ki-Ho Son. Effects of white led lighting with specific shorter blue and/or green wavelength on the growth and quality of two lettuce cultivars in a vertical farming system. *Agronomy*, 11(11), 2021. 1

[51] C.C.S. Nicole, J. Mooren, A.T. Pereira Terra, D.H. Larsen, E.J. Woltering, L.F.M. Marcelis, J. Verdonk, R. Schouten, and F. Troost. Effects of led lighting recipes on postharvest quality of leafy vegetables grown in a vertical farm. *Acta Hortic*, 1256:481–488, 2019. 1

[52] Franklin C. Ogidi, Mark G. Eramian, and Ian Stavness. Benchmarking Self-Supervised Contrastive Learning Methods for Image-Based Plant Phenotyping. *Plant Phenomics*, 5:0037, 2023. 3

[53] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 5

[54] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, A Antiga, and A Lerer. Automatic differentiation in pytorch. paszke2017automatic 5: 1–4. *DOI: https://doi. org/10.1145/3434309*, 2017. 7

[55] Michael P Pound, Jonathan A Atkinson, Alexandra J Townsend, Michael H Wilson, Marcus Griffiths, Aaron S Jackson, Adrian Bulat, Georgios Tzimiropoulos, Darren M Wells, Erik H Murchie, et al. Deep machine learning provides state-of-the-art performance in image-based plant phenotyping. *Gigascience*, 6(10):gix083, 2017. 2

[56] Harshadkumar B Prajapati, Jitesh P Shah, and Vipul K Dabhi. Detection and classification of rice plant diseases. *Intelligent Decision Technologies*, 11(3):357–373, 2017. 2

[57] Thararat Puangsuwan and Olarik Surinta. Enhancement of plant leaf disease classification based on snapshot ensemble convolutional neural network. 8

[58] Vinayakumar Ravi, Vasundhara Acharya, and Tuan D. Pham. Attention deep learning-based large-scale learning classi-

fier for cassava leaf disease classification. *Expert Systems*, 39(2):e12862, 2022. 8

[59] Brian D Ripley. The second-order analysis of stationary point processes. *Journal of applied probability*, 13(2):255–266, 1976. 6

[60] Hanno Scharr, Tony P Pridmore, and Sotirios A Tsaftaris. Computer vision problems in plant phenotyping, cvppp 2017–introduction to the cvppp 2017 workshop papers. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 2020–2021, 2017. 1

[61] Joao Paulo Schwarz Schuler, Santiago Romani, Mohamed Abdel-Nasser, Hatem Rashwan, and Domenec Puig. Grouped Pointwise Convolutions Reduce Parameters in Convolutional Neural Networks. *MENDEL*, 28(1):23–31, June 2022. Number: 1. 2

[62] Fereshteh Shahoveisi, Hamed Taheri Gorji, Seyedmojtaba Shahabi, Seyedali Hosseinirad, Samuel Markell, and Fartash Vasefi. Application of image processing and transfer learning for the detection of rust disease. *Scientific Reports*, 13(1):5133, 2023. 2

[63] Malleshaiah SharathKumar, Ep Heuvelink, and Leo F.M. Marcelis. Vertical farming: Moving from genetic to environmental modification. *Trends in Plant Science*, 25(8):724–727, 2020. 1

[64] Parul Sharma, Yash Paul Singh Berwal, and Wiqas Ghai. Performance analysis of deep learning cnn models for disease detection in plants using image segmentation. *Information Processing in Agriculture*, 7(4):566–574, 2020. 2

[65] Tingting Shi, Yongmin Liu, Xinying Zheng, Kui Hu, Hao Huang, Hanlin Liu, and Hongxu Huang. Recent advances in plant disease severity assessment using convolutional neural networks. *Scientific Reports*, 13(1):2336, 2023. 2

[66] Shigeharu Shimamura, Kenta Uehara, and Seiichi Koakutsu. Automatic identification of plant physiological disorders in plant factories with artificial light using convolutional neural networks. *International Journal of New Computer Architectures and Their Applications*, 9(1):25–31, 2019. 2

[67] Vimal Shrivastava, Monoj Pradhan, S. Minz, and M. Thakur. Rice plant disease classification using transfer learning of deep convolution neural network. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-3/W6:631–635, 07 2019. 2

[68] Davinder Singh, Naman Jain, Pranjali Jain, Pratik Kayal, Sudhakar Kumawat, and Nipun Batra. Plantdoc. *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD*, Jan 2020. 2

[69] Davinder Singh, Naman Jain, Pranjali Jain, Pratik Kayal, Sudhakar Kumawat, and Nipun Batra. Plantdoc: A dataset for visual plant disease detection. In *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD*, CoDS COMAD 2020, page 249–253, New York, NY, USA, 2020. Association for Computing Machinery. 2, 3, 7, 8

[70] Rahul Singh, Avinash Sharma, Neha Sharma, and Rupesh Gupta. Automatic detection of cassava leaf disease using transfer learning model. In *2022 6th International Conference on Electronics, Communication and Aerospace Technology*, pages 1135–1142, 2022. 8

[71] Srdjan Sladojevic, Marko Arsenovic, Andras Anderla, Dubravko Culibrk, Darko Stefanovic, and others. Deep neural networks based recognition of plant diseases by leaf image classification. *Computational intelligence and neuroscience*, 2016, 2016. Publisher: Hindawi. 2

[72] Ian Stavness, Valerio Giuffrida, and Hanno Scharr. Computer vision in plant phenotyping and agriculture. *Frontiers in Artificial Intelligence*, 6:1187301, 2023. 1

[73] Sharifah Farhana Syed-Ab-Rahman, Mohammad Hesam Hesamian, and Mukesh Prasad. Citrus disease detection and classification using end-to-end anchor-based deep learning model. *Applied Intelligence*, 52(1):927–938, 2022. 2

[74] Huy-Tan Thai, Kim-Hung Le, and Ngan Luu-Thuy Nguyen. Formerleaf: An efficient vision transformer for cassava leaf disease detection. *Computers and Electronics in Agriculture*, 204:107518, 2023. 8

[75] You-Wen Tian and Cheng-hua LI. Color image segmentation method based on statistical pattern recognition for plant disease diagnose. *Journal of Jilin University of Technology (Natural Science Edition)*, 2:028, 2004. 2

[76] Edna Chebet Too, Li Yujian, Sam Njuki, and Liu Yingchun. A comparative study of fine-tuning deep learning models for plant disease identification. *Computers and Electronics in Agriculture*, 161:272–279, 2019. Publisher: Elsevier. 2

[77] Dionysios Touliatos, Ian C. Dodd, and Martin McAinsh. Vertical farming increases lettuce yield per unit area compared to conventional horizontal hydroponics. *Food and Energy Security*, 5(3):184–191, 2016. 1

[78] Jindong Wang, Yiqiang Chen, Wenjie Feng, Han Yu, Meiyu Huang, and Qiang Yang. Transfer learning with dynamic distribution adaptation. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(1):1–25, 2020. 7

[79] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. pages 9929–9939. PMLR, 2020. 5

[80] Ross Wightman. Pytorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019. 8

[81] Hugh F Williamson, Julia Brettschneider, Mario Caccamo, Robert P Davey, Carole Goble, Paul J Kersey, Sean May, Richard J Morris, Richard Ostler, Tony Pridmore, et al. Data management challenges for artificial intelligence in plant and agricultural research. *F1000Research*, 10, 2021. 2

[82] Bichen Wu, Chenfeng Xu, Xiaoliang Dai, Alvin Wan, Peizhao Zhang, Zhicheng Yan, Masayoshi Tomizuka, Joseph Gonzalez, Kurt Keutzer, and Peter Vajda. Visual transformers: Token-based image representation and processing for computer vision, 2020. 8

[83] Harvey Wu, Tyr Wiesner-Hanks, Ethan L. Stewart, Chad DeChant, Nicholas Kaczmar, Michael A. Gore, Rebecca J. Nelson, and Hod Lipson. Autonomous Detection of Plant Disease Symptoms Directly from Aerial Imagery. *The Plant Phenome Journal*, 2(1):190006, 2019. 2

[84] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pages 12310–12320. PMLR, 2021. 3

[85] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 4

[86] Jing Zhang and Shuang-xi Wang. A study on the segmentation method in image processing for plant disease of gree nhouse [j]. *Journal of Inner Mongolia Agricultural University (Natural Science Edition)*, 3, 2007. 2

[87] Yiwei Zhong, Baojin Huang, and Chaowei Tang. Classification of cassava leaf disease based on a non-balanced dataset using transformer-embedded resnet. *Agriculture*, 12(9), 2022. 8

[88] Xin Zuo, Jiao Chu, Jifeng Shen, and Jun Sun. Multi-granularity feature aggregation with self-attention and spatial reasoning for fine-grained crop disease classification. *Agriculture*, 12(9), 2022. 8