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 ViTbase8, ViTsmall8, 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. 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.
We compare with other compelling datasets for plant health and stress and attaching di...

We present a large dataset on plant canopies exceeding 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 has 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. PlantCLEF2015 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
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.

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 1073x650 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 $(x, y)$, the learning objective is:

$$\mathcal{L} = - \sum_{i} y_i \log(H(E(x_i)))$$  \hspace{1cm} (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 base8 and ViT small8 [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 base8 and ViT small8 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:

- **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:

$$\hat{x} = \mu x_i + (1 - \mu) x_j$$
$$\hat{y} = \mu y_i + (1 - \mu) y_j$$  \hspace{1cm} (2)

with hyperparameter $\mu \in [0, 1]$.

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

<table>
<thead>
<tr>
<th>Encoder</th>
<th>Pre-trained on</th>
<th>Training Acc.</th>
<th>Validation Acc.</th>
<th>Test Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ViT small8</td>
<td>AGM*</td>
<td>0.9965</td>
<td>0.9720</td>
<td>0.9784</td>
</tr>
<tr>
<td>ViT small8</td>
<td>ImageNet</td>
<td>0.9305</td>
<td>0.9309</td>
<td>0.9298</td>
</tr>
<tr>
<td>ViT base8</td>
<td>AGM*</td>
<td>0.9505</td>
<td>0.9702</td>
<td>0.9798</td>
</tr>
<tr>
<td>ViT base8</td>
<td>ImageNet</td>
<td>0.9330</td>
<td>0.9303</td>
<td>0.9312</td>
</tr>
<tr>
<td>ResNet50</td>
<td>AGM*</td>
<td>0.9610</td>
<td>0.9440</td>
<td>0.9410</td>
</tr>
<tr>
<td>ResNet50</td>
<td>ImageNet</td>
<td>0.9512</td>
<td>0.9145</td>
<td>0.9243</td>
</tr>
</tbody>
</table>

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.

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 small8 model compared to milder augmentations. These findings emphasize the need...
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 meth-
ods rely on feature metrics based on the cosine-similarity
between two vectors of features $F_i, F_j \in \mathbb{R}^N$ [53]. For ex-
ample, with ViT the feature vectors could be the class token
cls, with $\text{cls} \in \mathbb{R}^{768}$, with smalls, and $\text{cls} \in \mathbb{R}^{1024}$ with ViT base. Cosine similarity compares pairs of feature vectors on a unit
hypersphere $S_{N-1}$ [79], image by image, but it cannot com-
pare sets of image features to study specific properties of
the induced feature space.

The purpose of this section is to show relevant differ-
ences 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 class-
ifier 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 coordi-
nates (for visualization purpose) from the feature average
pooling of ViT base 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 co-
nordinates of the features of the whole dataset, and each point
$p_j$ is the projection of three sampled coordinate features of
the $j$-th image.

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

$$
p_j = \left( \frac{r}{\|q_j\|} \right) q_j \tag{3}
$$

Note that because $w=0_k$ no translation is used in the map-
ing. Given $k$, the surface area of the hypersphere is:

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

**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
$D_{KL}(f_x||g)$, $x \in \{\text{ImageNet, AGM}\}$ with $f_x$ a kernel density
on the $k-1$ hypersphere, we evaluate how close the two dis-
tributions are from the uniform distribution $g \sim U(0, 1)$, see
Table 3. To make this meaningful in terms of spatial distri-
bution, 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 $\alpha$. We defined the spherical zones on the hy-
persphere as follows. Let $B_j$ be the regularized incomplete
beta-function, an hyperspherical cap $A_h$ with height $h$ is:

Let: $h = r - \alpha/2$ and $v = \left(\frac{2rh-h^2}{r^2}\right)$ then

$$
A_h = \frac{1}{2} S_{k} B_j \left( v; \frac{k-1}{2}, \frac{1}{2} \right) \tag{5}
$$
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 ViTBase8, 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:

$$S_{\text{zone}} = S_k - 2A_h$$  \hspace{1cm} (6)

For the kernel density estimation (KDE) [25], we used the Gaussian kernel over the points within the $k$ $S_{\text{zone,s}}$. 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 $\alpha$ 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 $\alpha$ on each zone surface. Since a cover of the caps with radius $\alpha$ 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 $S_{\text{zone}}$ the surface area and $A_h$ the surface of hypercaps with radius $\alpha$, an approximation of $K$, with $f$ the indicator function, and $\ell = r \cos^{-1}(p_ip_j/R^2)$ the arc length, is:

$$K(\alpha) = \frac{A_{\text{zone}}}{S_k} \sum_{i \neq j} I(\ell(p_i, p_j) < \alpha)$$  \hspace{1cm} (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.

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 $\text{AGM}_{\text{HS}}$, 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 $\text{AGM}_{\text{HS}}$ we also release the segmentation masks. This addition to the dataset enables the development and evaluation of advanced segmentation.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Zone</th>
<th># coord</th>
<th>divergence from $g \sim U(0,1)$</th>
<th>K-test</th>
</tr>
</thead>
<tbody>
<tr>
<td>AGM</td>
<td>Zone 1</td>
<td>4</td>
<td>2.5592</td>
<td>2.4701</td>
</tr>
<tr>
<td>ImageNet</td>
<td>Zone 1</td>
<td>4</td>
<td>0.1636</td>
<td>0.3804</td>
</tr>
<tr>
<td>AGM</td>
<td>Zone 2</td>
<td>4</td>
<td>2.6455</td>
<td>2.5498</td>
</tr>
<tr>
<td>ImageNet</td>
<td>Zone 2</td>
<td>4</td>
<td>0.1530</td>
<td>0.4436</td>
</tr>
<tr>
<td>AGM</td>
<td>Zone 3</td>
<td>4</td>
<td>2.7846</td>
<td>2.7118</td>
</tr>
<tr>
<td>ImageNet</td>
<td>Zone 3</td>
<td>4</td>
<td>0.1737</td>
<td>0.4275</td>
</tr>
<tr>
<td>AGM</td>
<td>Zone 4</td>
<td>4</td>
<td>2.6113</td>
<td>3.7714</td>
</tr>
<tr>
<td>ImageNet</td>
<td>Zone 4</td>
<td>4</td>
<td>0.1619</td>
<td>0.4360</td>
</tr>
</tbody>
</table>

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.
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.

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.

### 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.

<table>
<thead>
<tr>
<th>Encoder</th>
<th>Pre-trained on</th>
<th>Val. Acc (MLP)</th>
<th>Val. Acc (Linear)</th>
<th>Val. Acc (KNN)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ViT small$_8$</td>
<td>AGM</td>
<td>0.9720</td>
<td>0.8300</td>
<td>0.9502</td>
</tr>
<tr>
<td>ViT small$_8$</td>
<td>ImageNet</td>
<td>0.9477</td>
<td>0.9461</td>
<td>0.8702</td>
</tr>
<tr>
<td>ViT base$_8$</td>
<td>AGM</td>
<td>0.9273</td>
<td>0.9404</td>
<td>0.9102</td>
</tr>
<tr>
<td>ViT base$_8$</td>
<td>ImageNet</td>
<td>0.9078</td>
<td>0.8321</td>
<td>0.8989</td>
</tr>
<tr>
<td>ResNet50</td>
<td>AGM</td>
<td>0.9142</td>
<td>0.8800</td>
<td>0.8637</td>
</tr>
<tr>
<td>ResNet50</td>
<td>ImageNet</td>
<td>0.8625</td>
<td>0.8446</td>
<td>0.8230</td>
</tr>
</tbody>
</table>

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.

### 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.

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

Figure 9: Comparison of t-SNE 2-D projection of features from ImageNet and AGM pretrained models for crop $\times 1$ samples (above) and crop $\times 1$ samples (below), from AGM$_{HS}$.
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 Table 5. This table 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 $120 \times 120$, 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


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


[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


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


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


