MR. MARIO VALERIO GIUFFRIDA (Orcid ID : 0000-0002-5232-677X)

Article type : Technical Advance

Pheno-Deep Counter: a unified and versatile deep learning architecture for leaf counting

Mario Valerio Giuffrida [1,3], Peter Doerner[2], Sotirios A. Tsaftaris [1,4]

[1]Institute for Digital Communications, School of Engineering, University of Edinburgh, Edinburgh, UK

[2]School of Biological Sciences, University of Edinburgh, Edinburgh, Edinburgh, UK

[3] IMT School for Advanced Studies, Lucca, Italy

[4] The Alan Turing Institute, London, UK

Corresponding authors:

- Mario Valerio Giuffrida; v.giuffrida@ed.ac.uk
  Institute for Digital Communications, School of Engineering, University of Edinburgh
  Thomas Bayes Road, EH9 3FG, Edinburgh, UK
- Sotirios A. Tsaftaris; s.tsaftaris@ed.ac.uk
  Institute for Digital Communications, School of Engineering, University of Edinburgh
  Thomas Bayes Road, EH9 3FG, Edinburgh, UK, Tel: +44 (0)131 650 5796

Running Title: PhenoDC: a unified architecture for leaf counting

Keywords: Image-based plant phenotyping, machine learning, deep learning, leaf counting, multimodal, night images

Footnotes:

[1] M.V.G. implemented the method and analyzed data; M.V.G. and S.A.T planned the experiment; M.V.G., P.D. and S.A.T. wrote the article; all authors read and approved the article.

[2] This work was partially supported by the BBSRC grant BB/P023487/1.

[3] Corresponding authors: Mario Valerio Giuffrida (v.giuffrida@ed.ac.uk)

Abstract

Direct observation of morphological plant traits is tedious and a bottleneck for high-throughput phenotyping. Hence, interest in image-based analysis is increasing, requiring software that can reliably extract plant traits, such as leaf count, preferably across a variety of species and growth conditions. However, current leaf counting methods do not work across species or conditions and therefore may broad utility. In this paper, we present *Pheno-Deep Counter*, a single deep network that can predict leaf count in 2D plant images of different species with rosette-shaped appearance. We demonstrate that our architecture can count leaves from multi-modal 2D images, such as RGB, fluorescence, and near-infrared. Our network design is flexible, allowing for inputs to be added or removed to accommodate new modalities. Furthermore, our architecture can be used as is without requiring dataset-specific customization of the internal structure of the network, opening its use to new scenarios. *Pheno-Deep Counter* is able to produce accurate predictions in many plant species and, once trained, can count leaves in a few seconds. Through our universal and open source approach to deep counting, we aim to broaden utilization of machine learning based approaches to leaf counting. Our implementation can be downloaded at https://bitbucket.org/tuttoweb/pheno-deep-counter.

Introduction

Image-based plant phenotyping has recently become a valuable tool for quantitative analysis of plant images. However, its rapid expansion has highlighted the need for reliable software solutions, with the power to analyze data efficiently (Gehan et al., 2017). While previously the bottleneck was thought to be the acquisition of imaging data (*i.e.*, the hardware; Furbank and Tester, 2011), recently the bottleneck has shifted. The lack of reliable software (and algorithms) is currently creating a new bottleneck (Minervini et al., 2015a), due to the sheer amount of imaging data to be analyzed to extract quantitative plant traits. Machine learning has been proposed as a suitable solution to effectively extract plant traits (Singh et al., 2016; Tsaftaris et al., 2016).

Leaf count is an important plant trait and is directly related to the plants' development stage (Boyes et al., 2001), flowering time (Chien and Sussex, 1996), yield potential (Kouressy et al., 2006), and health (Rahnemoonfar and Sheppard, 2017). Until recently, leaf counting was treated as a by-product of leaf segmentation with deterministic image processing techniques. For example, most of the methods in the seminal collation study in leaf segmentation (Scharr et al., 2016), perform the following processing steps: first, they isolate the plant from the outer background (per-plant segmentation), then apply certain heuristics to delineate each leaf (per-leaf segmentation). For example, IPK (Pape and Klukas, 2015a), uses color images to extract geometrical representations of the isolated plant to find suitable split points to separate each leaf, relying on assumptions on plant shape and structure (*e.g.* reduced leaf overlap and visible long leaf blades). *Aksoy et al.* (2015) employ a clustering algorithm to delineate leaves on near-infrared images of Tobacco plants, where the per-leaf segmentation is further improved via shape models. In general, the main drawback of these deterministic approaches is that such heuristics may fail when encountering new data, reducing their applicability to different setups: for example, the performance of IPK drops by >20% when this algorithm is applied to Tobacco plants, where blade overlap is significant (Pape and Klukas, 2015a). Hence, users are faced with a dilemma: either adapt the many parameters of these methods or completely derive new *ad hoc* heuristic methods suitable to new imaging settings and plant species.

Machine learning is an alternative approach: rather than having users adapt methods, instead they provide their expertise by doing tasks they know well and have always been doing: phenotyping by observation. For example, in the context of leaf counting, a user may give observations (referred to as annotations in machine learning) either by delineating each leaf (a time-consuming finely-grained annotation), or by giving locations of each leaf (less time consuming), or just the total number of leaves in each plant. It is then the task of the machine learning algorithm to learn from such examples (known as the training set) *i.e.* the combination of images and corresponding annotations.

Romera-Paredes and Torr (2016) and *Ren and Zemel* (2017) proposed very sophisticated deep neural network models that, given a training set of images and precise leaf delineations, learn per-leaf segmentation and leaf count. They both evaluated their general method also in plant images of wild-type Arabidopsis based on an open dataset (Minervini et al., 2016). However, the collection of such finely-grained annotations is tedious and time-consuming, particularly when one must annotate data of significant diversity to account for large leaf variation, different imaging conditions, etc. In addition, these methods are very sensitive to how leaves are arranged (*i.e.,* plant topology). Due to the intricacies of the learned models, such approaches cannot fully accommodate the variability of leaf appearance and arrangement not seen during training.

Therefore, it remains of interest to identify methods that can learn robust leaf counting predictors without the need for such sophisticated annotations. *Giuffrida et al.* (2015) and *Pape and Klukas* (2015b) made the observation that elementary cues in the image could relate to plant leaf count. A predictor can thus be built by first extracting the cues (features) from images and then relating to the corresponding total leaf count. In particular, *Pape and Klukas* (2015b) used hand-designed geometric features from the per-plant segmentation mask to learn a relationship (a regression) between such features and leaf count. This approach required expert knowledge of appropriate geometric features to use. On the contrary, the method of *Giuffrida et al.* (2015), uses K-means (Coates et al., 2011), instead of hand-designed features, to learn a visual dictionary from the data in a context adaptive fashion without expert knowledge.

Recently, deep neural networks have also been employed to address the leaf counting problem. These approaches essentially combine the task of finding suitable image features with the task of learning a good regression model relating the features to leaf count (Aich and Stavness, 2017; Dobrescu et al., 2017a; Ubbens and Stavness, 2017). These approaches show significant promise, but each of these is specialized: a new model and network for each plant species or cultivar, imaging condition, *etc* is required. In addition, all three approaches use only optical images, whereas different imaging sensors such as near-infrared or fluorescence are now also commonly employed in plant phenotyping (Alpet et al., 2015; Fiorani and Schurr, 2013; Gehan et al., 2017; Klukas et al., 2014).

In this paper, we introduce the *Pheno-Deep Counter* (shorthanded as *PhenoDC*), a multi-input deep network that combines information coming from different imaging sources (termed *modalities* hereafter) to count the number of leaves of rosette-shaped plants. In contrast to other approaches, we aim to build a single unified model that can be used for a variety of plants and imaging scenarios, where plants are seen from the top in a laboratory setting. Critically, we demonstrate that, by agglomerating data from a variety of sources, the model learns better (deep learning algorithms require large amounts of data; Sun et al., 2017). Our approach also significantly enhances utility, as the same model can be used in a variety of scenarios and can be easily adapted for this purpose.

The main contributions of this work are:

1. **Multi-modal model:** an architecture that benefits from, and can use, multiple imaging modalities, *e.g.* classical color (RGB) *and* near-infrared images. We show that by combining information coming from multiple modalities, *PhenoDC* improves leaf count prediction. As an example, training our network with only RGB images, *PhenoDC* predicts the correct leaf count in 55% of the cases. Adding other modalities (e.g., near-infrared and fluorescence), the prediction accuracy increases to 88%.
2. **Ease of adaptation to new settings:** our model can be easily adapted to work with another imaging setup (still assuming top-view), either by simply specializing the network to the new task or performing data agglomeration. We show that with a handful plant images (regardless of the species tested), our network can be trained to count leaves for the new scenario. We showcase several experiments using images of *Arabidopsis thaliana* plants, as well as other plant species, such as Tobacco and Komatsuna (a Japanese vegetable).
3. **State-of-the-art performance:** our approach can predict the number of leaves in unseen images with an error of ±1 leaf in ~80% of the cases as compared to 57% in *Giuffrida et al.* (2015) closing further the gap in achieving human-level performance (Giuffrida et al., 2018). This improves further when multi-modal learning is used.
4. **Nocturnal leaf counting:** we show that our network is also capable of counting leaves during the night with near-infrared images, extending the applicability throughout the diel cycle, a feature not yet addressed by any other methods.

We perform a comprehensive analysis and comparison with other methods using a variety of data sources (both in-house and publicly available). To aid adoption of our approach, we release code and trained models to allow the plant community to utilize them in their experiments. This work also includes several experiments and discussion points to help elucidate how one can adopt such approach (*e.g.* how many annotated samples are required and how to collect annotations) and how to interpret findings.

Results

To showcase the performance of our approach, we employed four different datasets:

1. A special collection of the PRL dataset (Minervini et al., 2016) and Aberystwyth dataset (Bell and Dee, 2016) that have been used in the latest CVPPP 2017 *Leaf Counting Challenge (LCC)*[1]; it contains five different sub-datasets (*c.f.* Table S1). These datasets contain RGB color images of four different plant experiments, using different plants (and different cultivars), growth conditions, and camera settings.
2. The multi-modality imagery database for plant phenotyping (Cruz et al., 2015), containing images of Arabidopsis thaliana Col-0 acquired in three different modalities (RGB, near-infrared, fluorescence);
3. The RGB images in Komatsuna dataset (Uchiyama et al., 2017);
4. Nocturnal Arabidopsis plant images acquired using a near-infrared camera (Dobrescu et al., 2017b).

Visual samples of these datasets are shown in Figure S1, whereas technical details are reported in Table S1.

Our deep neural network, shown in Figure 1 and detailed in *Experimental setup*, has been designed with the aim to accommodate inputs of variable size. To achieve this, our architecture breaks down the task of counting into several sub-tasks. First, each image goes through a network that aims to find a fixed length vector representation to better describe a plant image. This is achieved by a sub-network (*modality branch)*, where each input source is processed independently. However, during training the network learns what is useful to retain from each modality, which results in an image descriptor (a vector per image) that jointly represents all the useful information. Multi-modal plant representation is accomplished by the *feature fusion* part of the architecture (details in *Experimental setup*). Finally, the fused image descriptor is related to leaf count, by learning the parameters of a non-linear regression model between the descriptor and leaf count. After the network has been trained, evaluation of a

[1] More information at: https://www.plant-phenotyping.org/CVPPP2017-challenge

plant's image(s) (the plural is used to denote the presence of different modalities) provides an estimate of the leaf count.

To quantitatively assess the performance of our approach, we adopt the same evaluation metrics as in *Giuffrida et al.* (2015) (now a consensus in the broad community):

- *Difference in Count* (DiC): mean and standard deviation of the differences between predicted leaf counts and ground-truth (best value when mean and standard deviation are close to 0);
- *Absolute Difference in Count* (|DiC|): similar as before, but the differences between prediction and ground-truth are in absolute value (best value when mean and standard deviation are close to 0);
- *Mean Squared Error* (MSE): mean of the squared differences between prediction and ground-truth (best value near to 0);
- *Percentage Agreement* (%): number of times (as a percentage) that the predicted leaf count is exactly correct (best value at 100%).

Technical details about our deep architecture are provided in the *Experimental setup* section, whereas evaluation metrics are detailed in Supplemental Methods S1.

We present a comprehensive set of experiments that demonstrate the reliability of *PhenoDC* for leaf counting. To train our model, data are split into (at least) two datasets, namely training and testing set. The training set is needed to optimize the set of parameters specifying our model (*c.f. Experimental procedures* section for further details). The testing set is required to evaluate the performance of the algorithm, using *unseen* data.

In the following, in a series of experiments we show:

(a) the benefit of data agglomeration across different sources;
(b) the superior prediction performance in the recent benchmark *CVPPP 2017* dataset;
(c) that prediction error reduces when using multimodal sources using the dataset of *Cruz et al.* (2015); and lastly
(d) a set of experiments that demonstrate the flexibility of our network to adapt to other contexts, such as different plant species.

Proof-of-concept: data agglomeration helps

Herein, we aim to show that increasing data diversity in fact improves accuracy.

We isolated the A1 set of images in the *CVPPP 2017* dataset (Minervini et al., 2016), which includes 128 images of *Arabidopsis thaliana* Col-0 for training. We followed the training procedure of *Dobrescu et al.* (2017a), assessing the performance of our network using a 4-fold cross-validation, splitting randomly the training set with the following proportions: (i) 64 images for learning; (ii) 32 images for validation; and (iii) 32 images for testing. The validation set allows us to monitor model performance during training and prevents overfitting (the case where the model has essentially memorized the training set and therefore cannot adapt to new data).

Using this learning protocol, the 4-fold cross-validation results are the following:

- *DiC*: -0.81 (0.85);
- |*DiC*|: 0.94 (0.70);
- *MSE*: 1.38;
- *Percentage Agreement:* 25%.

We proceeded to add more data drawn from the CVPPP 2017 dataset, namely the A2 (*Arabidopsis thaliana* of 5 genotypes), A3 (Tobacco), and A4 (*Arabidopsis thaliana* Col-0) set of images. As we progressed adding data, we observed that the mean squared error reduced by ~50% (MSE: 0.72). A similar improvement was seen in the *Percentage Agreement*, which increased to 56%. Finally, we wanted to evaluate which areas of an image contribute to the count. Ideally, the count produced by the network should only be influenced by regions of the image containing plant. This analysis was performed using the method in *Dobrescu et al.* (2017a). We describe this analysis in Supplemental Methods S2 and show evaluation in Figure S2 on sample images taken from the *CVPPP 2017* dataset.

This experiment highlights the benefit of data agglomeration, even when the sources are diverse. Since deep networks can form very complex functions (between input and output) more data is better and being "universal" is better than being specialized (for example one model per plant species) as it reduces the chance of memorization.

Evaluation and comparison with state-of-the-art on the CVPPP 2017 benchmark dataset

In this experiment, we assess the performance of our network when trained on the heterogeneous CVPPP 2017 plant dataset and how it compares to state-of-the-art methods in the literature.

We report quantitative results in Table 1, comparing our performance with other deep learning methods for leaf counting (Aich and Stavness, 2017) and leaf counting via segmentation (Romera-Paredes and Torr, 2016; Ren and Zemel, 2017), as well as with the machine learning algorithm that won *CVPPP LCC 2015* (Giuffrida et al., 2015). The CVPPP 2017 dataset contains as a subset data of previous competitions allowing comparisons across the years and methods (but not on all data). Overall, PhenoDC outperforms all others, scoring the lowest MSE error in all datasets (1.56). Note that the single input model of our deep architecture achieved the best results on the *CVPPP 2017* dataset in the *Leaf Counting Challenge* (LCC). A paired t-test shows statistically significant gains when compared to *Aich and Stavness, 2017* (p-value < 0.0001; last column of Table 1). Figure 2 collates results across all images as: (a) the correlation between ground-truth and prediction, showing high agreement of our method (R^2 = 0.96); (b) the distribution of error in leaf count, where it can be seen that in ~80% of the cases the error is confined within the ±1 leaf range (for comparison *Giuffrida et al. (2015)*, report 57% agreement on the same range).[2] In some occasions, *PhenoDC* might predict leaf counts incorrectly. In Figure S3, we show some examples of such cases taken from the training set. (Figure S3a: ground-truth 20, predicted 17; Figure S3b: ground-truth 18, predicted: 15; Figure S3c: ground-truth: 13, predicted: 7). Overall, these images show several challenges to the network, including significant overlap and concentrated small leaves in the central part of the plant.

In conclusion, *PhenoDC* is more reliable in terms of leaf counting, compared to the current state of the art approaches.

[2] It is relevant to point out that, differently from our method, *Giuffrida et al.* (2015) used only the A1, A2, and A3 images. *PhenoDC* still has an accuracy of ±1 leaf range ~80%, when trained and tested on the same portion of the data to make fair comparisons.

Multiple modalities and leaf counting

In this section, we assess whether our network benefits from multi-modal learning, leading to improved leaf count predictions.

For this experiment, we used the dataset of *Cruz et al.* (2015), which contains images of *Arabidopsis thaliana* wild-type (*Col-0*) acquired using multiple sensors. Cruz and collaborators used 16 plants for 9 days, acquiring top-view images from 9am to 11pm (15 frames a day). This setup produced a dataset containing 2,160 individual images altogether, albeit only 576 images are annotated (images taken at 9am, 12pm, 4pm, and 8pm). Images were taken simultaneously in the following modalities: visible light (RGB), fluorescence (FMP), near-infrared (NIR), and depth. The multiple sensors acquired the same plants simultaneously. Due to the heterogeneity of such sensors and their placement, image resolution (and effective image size) and alignment vary. We excluded depth images due to their extremely low resolution (~30×30 pixels), compared with the others (*c.f.* Table S1). Image samples are shown in Figure S1 with dataset details reported in Table S1. We randomly split the labeled dataset into 3 parts (50% training, 25% validation, and 25% testing) and trained our network using 4-fold cross-validation.

To establish a baseline for our multi-modal results and to find the most useful single modality (for the counting task), we first trained our network using only one of the available modalities as input at a time prior to using all modalities. As reported in Table 2, we obtained the best single-input result using the near-infrared (NIR) images (MSE = 0.39). This is due to the fact that NIR images, in this dataset, are sharper and more detailed. To demonstrate this, we visualize the activations produced by our network for each of the modality branches. In Figure S4, we show the output of the first residual block (He et al., 2016) for three sample plants of the dataset (mean activation across the feature maps). Overall, most of the activations are focused on the region where the plant is located. Note that, while some pixels are active on the background on RGB or FMP, the IR activations are mostly dominant on the plant, which demonstrate the benefit of using multi-modal information. We obtained the best performance when all three inputs were used simultaneously: MSE was reduced by more than 50%, and *Percent Agreement* increased by ~19%.

We conclude that combining information coming from multiple modalities improves counting accuracy. The fusion layer learns (*c.f.* Figure 1b) to retain the most useful image features coming from any of the modality branches (*c.f.* Figure 1c). These experiments highlight that multi-modal learning can be useful for plant phenotyping purposes, and that our architecture can handle any number of inputs.

Evaluation of network adaptivity capabilities

In this section, we address the problem of how one can use *PhenoDC* by adapting to other experimental setups, different to the one used during training.

We rely on the principle of fine-tuning a *pre-trained* network to significantly reduce the number of new training examples required to adapt the network (Bengio, 2012) and increase performance (Sun et al., 2017). Fine-tuning entails the labeling of *just* few images and their use to update the parameters of a network that has been pre-trained to solve the same task but in a different context (*e.g.* different plant species).

We demonstrate this capability in three different cases using the following datasets: Tobacco plants (A3) from *Minervini et al.* (2016), the Komatsuna plants from *Uchiyama et al.* (2017), and other Arabidopsis cultivars using night-time images *Dobrescu et al.* (2017b). (Further details of all these image datasets are in Table S1 and Figure S1). For these experiments, we first pre-trained our neural network using only the Arabidopsis plant images A1, A2, and A4, in the *CVPPP 2017* dataset (Bell and Dee, 2016; Minervini et al., 2016). This training dataset containing Arabidopsis plants, as reported in Table S1, does not contain a large number of images, making the learning process challenging. The following experiments also were aimed to assess the number of training images required to adapt the network into another scenario.

Tobacco plants [different species, imaging camera, and settings]: We fine tuned the pre-trained network using a variable number of tobacco training images. Specifically, we selected 7, 14, 21, and then 27 images to fine-tune the pre-trained network. The results of these experiments are reported in Table 3. Overall, we observe that more training data leads to better predictions in the testing set. As expected, the lowest error is obtained when we use all 27 images for training (MSE = 1.50). In Figure 3, we show the distribution of the error that we registered during progressive learning. As more images are used, the error distribution narrows around the 0. In fact, in ~80% of the data in the testing set our method is within 1 leaf error from the ground-truth (highlighted areas in Figure 3), thus achieving more accurate predictions. Hence, we can conclude that fine-tuning with a handful of images (>=21 in this setup), *PhenoDC* can produce reliable leaf count.

The Komatsuna case [different species, imaging camera, and settings]: this dataset contains 300 RGB images of 5 different Komatsuna plants, 6 images/day for 10 days. (Images were taken from 3pm until 3pm every 4 hours). We split the dataset as follows (*c.f.* Table S1):

- *training set*: 2 plants (IDs 00 and 01), entire timeline (120 images);
- *validation set*: 1 plants (ID 04), entire timeline (60 images);
- *testing set*: 2 plants (IDs 02 and 03), entire timeline (120 images).

We fine-tuned our pre-trained network by progressively increasing the training set size to 10, 20, 30, and then 40 images per plant, choosing time frames that followed the plant growth evolution. Overall, the results in Table 4 show that more data contribute to more accurate results. Predictions become very accurate when 40 images per plant are used during training, showing a reduction of the MSE by 50%, compared with training using 10 images per plant.

Nocturnal images of Arabidopsis plants [different cultivars, settings and modality]: Night images are usually acquired using infrared cameras and specific LED lights that illuminate the scene with near-infrared radiation (wavelength of 940nm which does not alter natural plant development; Cruz et al., 2015; Dobrescu et al., 2017b). We selected and annotated a subset of night images (from *Dobrescu et al., 2017b*). Specifically, we selected 18 plants and sampled one image per night every other day for eight days (totally 72 images). Examples of nocturnal images are shown in Figure S1. We pre-trained the network using the NIR images from *Cruz et al.* (2015) and fine-tuned using 10 plants for training (40 images in total), 4 plants for validation (16 images), and the last 4 for testing (16 images). Since these images come from different ascensions of *A. thaliana*, we randomly changed the training/validation/testing set 4 times. Quantitative results on the testing error are: *DiC*: -0.14 (0.77); |*DiC*|: 0.52 (0.59); *MSE*: 0.61; and *Percent Agreement:* 53.1%. Overall, the error is very low (MSE < 1), demonstrating the utility of our machine learning approach to leaf counting during the night.

To summarize, these experiments demonstrated that *PhenoDC* can adapt to different scenarios of considerable complexity. Acceptable performance can be attained using few images (*e.g.* 14 in the case of Tobacco). In addition, by fine-tuning our network with Arabidopsis images acquired during the night, we permit plant growth analysis during the entire circadian cycle (Apelt et al., 2015).

Discussion

In this paper, we report *Pheno-Deep Counter*, a deep artificial neural network that can predict the total number of leaves from top-view plant images. We showed the effectiveness and reliability of our network architecture using several plant datasets. Specifically, we show that data agglomeration helps to improve accuracy: as more datasets were added, the mean squared error fell by 50%. A similar error reduction was also observed when the network was trained with multi-modal data, showing that combining information coming from multiple imaging sources helps to train a better regression model and to learn better features. We showed that our method can adapt to new settings and demonstrated that a refinement step, fine-tuning, can be used to achieve excellent performance even with only a few images for training. We also demonstrate that NIR modalities can be used to count leaves during darkness, permitting leaf counts for detailed plant growth analysis throughout the circadian cycle.

Our approach to leaf counting learns meaningful image features across all modalities and then relates features to leaf count via non-linear regression. We train both aspects together, thus adapting image features whilst learning the regressor. This has been central to the success of deep learning in a variety of problems, from image recognition to self-driving cars (LeCun et al., 2015). Furthermore, our approach offers a single model to solve the same task for any input. Our robust and accurate neural network can be extended for new input/modalities without changing the overall architecture. This simplifies adoption and permits the sharing of model updates when new experiments have been made available on the basis of our architecture. Therefore, by placing our pre-trained *PhenoDC* and source code (and instructions) into the pubicly available at `https://bitbucket.org/tuttoweb/pheno-deep-counter`, we hope to accelerate the adoption of such methods in plant phenotyping analysis.

Our network was evaluated on top-down views of dicot rosette-shaped plants. Clearly, this is one setup, among many others. It is possible though that an *ideal* leaf counting algorithm would be able to work also on monocots, and even tree canopies with thousands of leaves, given enough training data. Unfortunately, we presently lack such curated datasets with these scenarios and we are unable to experimentally assess how *PhenoDC* would perform, albeit it still brings us a step closer towards generalization.

In this work, we focused on "*how many*", rather than "*which*" annotated images, are needed to train a good regression model. It goes without saying that adequate image resolution and quality are necessary. Generally, images that show appearance diversity are good images to annotate. In a time-lapse setting, images spanning a set interval of the time series would be a good start. However, better approaches exist to find the best set of images to jointly inform the model, known as active learning. Active learning with neural networks is an ongoing research problem in machine learning. We previously showed, using plant descriptors and data mining (He, 2016), a promising potential in identifying images for annotation.

Furthermore, this work assumed that ground-truth annotations (provided by expert observers) are considered as gold standard and error-free. However, it is widely known (*e.g.*, in medical image analysis applications) that even expert observers show variation. Recently, several related works showed that variations exist among annotators in labeling plant images (Giuffrida et al., 2018), or in assigning specific (a)biotic plant stress via visual inspection of leaf blades (Ghosal et al., 2018). Interestingly, intra- and inter-observer variation can be used to also assess algorithm performance. Based on the findings of *Giuffrida et al.* (2018), inter-observer variation has a mean square error of 0.81 (non-experienced annotators on a subset of Arabidopsis images used in *Minervini et al., 2017*). Experienced and non-experienced annotators are within the ±1 leaf error range in the ~90% of the cases, whereas *PhenoDC* is within ±1 leaf error in ~80% of the cases, thus bringing us closer to human-level performance.

Evidently, "*true*" ground-truth data can only be attained by aggregating observations from many annotators to reach a consensus. Since doing so with experts is time-consuming, recent studies using online dedicated platforms, such as *Zooniverse*, can alleviate this problem, by tapping into the power of citizen scientists. An alternative is to use simulated or synthetic data, where ground-truth is absolute by design. Simulated data have recently been used in the plant community to count the number of fruits (Rahnemoonfar and Sheppard, 2017) and the number of leaves in Arabidopsis plants (Mündermann et al., 2015; Ubbens et al., 2018). Simulated images are provided by a software that takes object parameters as input (*e.g.*, plant age, number of leaves). However, images may lack visual realism, but recent innovations in image synthesis (Giuffrida et al., 2017) point to the potential of creating synthetic images of realistic appearance.

In conclusion, we present a deep learning approach to leaf counting with a neural network. Trained with examples of images and corresponding plant leaf counts, our approach can achieve outstanding results in a variety of settings. Our model handles many input modalities and has been tested with images of different species, cultivars and also with images at night. By making it openly available to the community we hope that it will stimulate large-scale analysis in plant phenotyping of a crucial plant trait: leaf count and help relieve the analysis bottleneck (Minervini et al., 2015a).

Experimental procedures

In this section, we discuss technical details of the deep network architecture characterizing our *Pheno-Deep Counter*, shown in Figure 1. We optimize all computational blocks simultaneously to obtain a mapping between input images and leaf counts. For our purposes, we used up to three inputs: RGB, near-infrared, and fluorescence.

Modality branch: The sub-network that processes each input (*c.f.* Figure 1a). We used the *ResNet50* architecture (He et al., 2016), as in (Dobrescu et al., 2017a). Each input is processed independently from others and generates a vector representation specific to its input, ensuring meaningful and discriminative features. Each branch ends with a fully connected layer of 1,024 neurons using a rectifier (ReLU) non-linearity, which allows the suppression of negative values during the process of feature extraction. Each input results in an output vector of the same size independent of input image size.

Feature Fusion: The process that combines information coming from all modalities to retain the most meaningful features. Following the concept in *Chartsias et al.* (2018), we apply an element-wise max fusion layer. We display this segment of the network in Figure 1b.

Regression: The process of relating fused information to leaf count (*c.f.* Figure 1c). The output of the fusion layer is given to another fully connected layer of 512 neurons with ReLU activation function. At the end of the network, the output of the last layer is given to a single neuron that makes the actual prediction of the number of leaves. During training, we minimize the mean squared error (MSE) between predicted leaf count and ground-truth. The model predicts real numbers and we round the leaf count to the nearest integer only at test time.

Training strategies: We employ three common training strategies to improve network training and performance. First, we initialise our network with pre-trained parameters (rather than random ones), computed previously based on image recognition task (Russakovsky et al., 2015). Second, we use an L_2 regularizer in the last fully connected layer before the output (*i.e.* the regression component). This technique prevents the network from learning large weights which may produce unstable results. For all experiments in this paper, we set this regularization constant to $\lambda=0.02$. Finally, to artificially increase robustness to view changes (rotation, translation and position of camera), we perform *dataset augmentation* during training. Specifically, we apply random geometrical transformations to the training data (*e.g.*, random rotations, zoom-ins, shifts). This helps the network to learn from more data without having to collect more data. Our network was trained using a learning rate of $\eta=0.0001$.

Validation set: One of the problems arising in network training is when to stop training. The typical approach in machine learning is to also use a small set of labeled data, called the validation set (Theodoridis and Koutroumbas, 2008). We therefore used an early stop criterion to interrupt the learning procedure, terminating the training after 10 epochs we observe that the validations error has started to get worse.

Image Preprocessing: While combining data across different sources (data agglomeration) has benefits, the images coming from different setups exhibit variations in intensity and size that need to be corrected. For instance, images in A1 (Minervini et al., 2016) and images in A4 (Bell and Dee, 2016) were acquired with different cameras and different illumination conditions, although they may show the same plant species (*Arabidopsis thaliana Col-0*). To ameliorate variations in illumination, we perform histogram normalization on all images and to standardize image size we resize all images of a modality to the same size 320×320 pixels. For the multi-modal images (Cruz et al., 2016), RGB images are too small to be provided to the RGB modality branch, as ResNet needs images at least of 200x200 pixels size (*c.f.*, Table

S1). In this case, we upsampled the images to 240×240 pixels, whereas the images from the other modalities were left unchanged.

Implementation details: We implemented our deep neural network using *Keras* (Chollet, 2015), an open-source library for deep learning in python, with Tensorflow backend. We performed our training experiments in a machine endowed with a TITAN X GPU. Note that such equipment is not necessary for fine-tuning and adapting our network to new experimental data.

Acknowledgments

We thank Nvidia Corp. for providing the GPU used for this paper. We also thank Andrei Dobrescu for his valuable help.

Conflict of interest

The authors declare no conflict of interest.

Supplemental Material

Supplemental Figure S1. Sample images of the employed datasets.

Supplemental Figure S2. Visualization of which part of an image contributes for the leaf counting.

Supplemental Figure S3. Some examples of images taken from the CVPPP dataset where the leaf count prediction is inaccurate.

Supplemental Figure S4. Visualization of the output of the first residual block for each of the modality branches.

Supplemental Table S1. Details of the plant phenotyping datasets used in this paper.

Supplemental Methods S1. Evaluation metrics

Supplemental Methods S2. Assessing what the network counts

References

Aich, S., and Stavness, I. (2017). Leaf Counting with Deep Convolutional and Deconvolutional Networks. In IEEE International Conference on Computer Vision Workshops Workshops (pp. 2080–2089). IEEE Computer Society.

Aksoy, E. E., Abramov, A., Wörgötter, F., Scharr, H., Fischbach, A., and Dellen, B. (2015). Modeling leaf growth of rosette plants using infrared stereo image sequences. Computers and Electronics in Agriculture, 110, 78–90.

Apelt, F., Breuer, D., Nikoloski, Z., Stitt, M., and Kragler, F. (2015). Phytotyping 4D : A light-field imaging system for non-invasive and accurate monitoring of spatio-temporal plant growth. The Plant Journal, 82(4), 693–706.

Bell, J., and Dee, H. (2016). Aberystwyth Leaf Evaluation Dataset. URL: https://doi.org/10.5281/zenodo.168158.

Bengio, Y. (2012). Deep Learning of Representations for Unsupervised and Transfer Learning. In Proceedings of ICML Workshop on Unsupervised and Transfer Learning (Vol. 27, pp. 17–36).

Boyes, D. C., Zayed, A. M., Ascenzi, R., McCaskill, A. J., Hoffman, N. E., Davis, K. R., and Görlach, J. (2001). Growth stage-based phenotypic analysis of Arabidopsis: a model for high throughput functional genomics in plants. The Plant Cell, 13(7), 1499–510.

Chartsias, A., Joyce, T., Giuffrida, M. V., and Tsaftaris, S. A. (2017). Multimodal MR Synthesis via Modality-Invariant Latent Representation. IEEE Transactions on Medical Imaging, vol. 37, no. 3, pp. 803–814.

Chien, J. C., and Sussex, I. M. (1996). Differential Regulation of Trichome Formation on the Adaxial and Abaxial Leaf Surfaces by Gibberellins and Photoperiod in Arabidopsis thaliana (L.) Heynh. Plant Physiology, 111(4), 1321–1328.

Chollet, F. (2015). Keras. GitHub repository: https://github.com/fchollet/keras

Coates, A., Arbor, A., and Ng, A. Y. (2011). An Analysis of Single-Layer Networks in Unsupervised Feature Learning. *Aistats 2011*, 215–223.

Cruz, J. A., Yin, X., Liu, X., Imran, S. M., Morris, D. D., Kramer, D. M., and Chen, J. (2016). Multi-modality imagery database for plant phenotyping. Machine Vision and Applications, 27(5), 735–749.

Dobrescu, A., Giuffrida, M. V., and Tsaftaris, S. A. (2017a). Leveraging multiple datasets for deep leaf counting. In the IEEE International Conference on Computer Vision Workshops (ICCVW), 2072–2079.

Dobrescu, A., Scorza, L. C. T., Tsaftaris, S. A., and McCormick, A. J. (2017b). A "Do-It-Yourself'' phenotyping system: measuring growth and morphology throughout the diel cycle in rosette shaped plants. Plant Methods, 13(1), 95.

Furbank, R. T., and Tester, M. (2011). Phenomics – technologies to relieve the phenotyping bottleneck. Trends in Plant Science, 16(12), 635–644.

Gehan, M. A., Fahlgren, N., Abbasi, A., Berry, J. C., Callen, S. T., Chavez, L. et al. (2017). PlantCV v2: Image analysis software for high-throughput plant phenotyping. PeerJ, 5, e4088.

Ghosal, S., Blystone, D., Singh, A. K., Ganapathysubramanian, B., Singh, A., and Sarkar, S. (2018). An explainable deep machine vision framework for plant stress phenotyping. Proceedings of the National Academy of Sciences of the United States of Americ*a*, *115*(18), 4613–4618.

Giuffrida, M. V., Minervini, M., and Tsaftaris, S. A. (2015). Learning to Count Leaves in Rosette Plants. In Proceedings of the Computer Vision Problems in Plant Phenotyping (CVPPP), pages 1.1–1.13. BMVA Press.

Giuffrida, M. V., Scharr, H., and Tsaftaris, S. A. (2017). ARIGAN: Synthetic Arabidopsis Plants using Generative Adversarial Network. In the IEEE International Conference on Computer Vision Workshops (ICCVW), 2064–2071.

Giuffrida, M. V., Chen, F., Scharr, H., and Tsaftaris, S. A. (2018). Citizen crowds and experts: Observer variability in image-based plant phenotyping. Plant Methods, 14(1).

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770–778.

He, S. (2016). Active learning for image processing, M.Sc. Thesis, The University of Edinburgh.

Klukas, C., Chen, D., and Pape, J.-M. (2014). IAP: an open-source information system for high-throughput plant phenotyping. Plant Physiology, 165(June), 506–518.

Kouressy, M., Dingkuhn, M., Vaksmann, M., Clément-Vidal, A., and Chantereau, J. (2008). Potential contribution of dwarf and leaf longevity traits to yield improvement in photoperiod sensitive sorghum. European Journal of Agronomy, 28(3), 195–209.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. Nature, 521(7553), 436–444.

Minervini, M., Scharr, H., and Tsaftaris, S. A. (2015a). Image analysis: The new bottleneck in plant phenotyping [applications corner]. IEEE Signal Processing Magazine, 32(4), 126–131.

Minervini, M., Giuffrida, M. V., and Tsaftaris, S. A. (2015b). An interactive tool for semi-automated leaf annotation. In Proceedings of the Computer Vision Problems in Plant Phenotyping (CVPPP), pages 6.1–6.13. BMVA Press.

Minervini, M., Fischbach, A., Scharr, H., and Tsaftaris, S. A. (2016). Finely-grained annotated datasets for image-based plant phenotyping. Pattern Recognition Letters, 81, 80–89.

Minervini, M., Giuffrida, M. V., Perata, P., and Tsaftaris, S. A. (2017). Phenotiki: an open software and hardware platform for affordable and easy image-based phenotyping of rosette-shaped plants. Plant Journal, 90(1), 204–216.

Mündermann, L., Erasmus, Y., Lane, B., Coen, E., and Prusinkiewicz, P. (2005). Quantitative modeling of Arabidopsis development. Plant Physiology, 139(2), 960–968.

Pape, J. M., and Klukas, C. (2015a). 3-D histogram-based segmentation and leaf detection for rosette plants. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 8928, 61–74.

Pape, J. M., and Klukas, C. (2015b). Utilizing machine learning approaches to improve the prediction of leaf counts and individual leaf segmentation of rosette plant images. In Proceedings of the Computer Vision Problems in Plant Phenotyping (CVPPP), pages 3.1–3.12. BMVA Press.

Rahnemoonfar, M., Sheppard, C. (2017). Deep Count: Fruit Counting Based on Deep Simulated Learning. Sensors, 17(4), 905.

Ren, M., Zemel, R. S. (2017). End-To-End Instance Segmentation With Recurrent Attention. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 293–301.

Romera-Paredes, B., and Torr, P. H. S. (2016). Recurrent instance segmentation. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) (Vol. 9910 LNCS, pp. 312–329).

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S. et al. (2015). ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision, 115(3), 211–252.

Scharr, H., Minervini, M., French, A. P., Klukas, C., Kramer, D. M., Liu, X., et al. (2016). Leaf segmentation in plant phenotyping: a collation study. Machine Vision and Applications, 27(4), 585–606.

Scharr, H., Minervini, Fischbach, A., Tsaftaris, S. A. (2014). Annotated Image Datasets of Rosette Plants. Technical Report No. FZJ-2014-03837, Forschungszentrum Jülich.

Singh, A., Ganapathysubramanian, B., Singh, A. K., Sarkar, S. (2016). Machine Learning for High-Throughput Stress Phenotyping in Plants. Trends in Plant Science, 21(2), 110–124.

Sun, C., Shrivastava, A., Singh, S., and Gupta, A. (2017). Revisiting Unreasonable Effectiveness of Data in Deep Learning Era. In Proceedings of the IEEE International Conference on Computer Vision (Vol. 2017–October, pp. 843–852).

Theodoridis, S., Koutroumbas, K. (2008). Pattern Recognition (Forth, Vol. 11). Elsevier B.V.

Tsaftaris, S. A., Minervini, M., and Scharr, H. (2016). Machine Learning for Plant Phenotyping Needs Image Processing. Trends in Plant Science, 21(12), 989–991.

Ubbens, J. R., and Stavness, I. (2017). Deep Plant Phenomics: A Deep Learning Platform for Complex Plant Phenotyping Tasks. Frontiers in Plant Science, 8.

Ubbens, J., Cieslak, M., Prusinkiewicz, P., and Stavness, I. (2018). The use of plant models in deep learning: An application to leaf counting in rosette plants. Plant Methods, 14(1).

Uchiyama, H., Sakurai, S., Mishima, M., Arita, D., Okayasu, T., Shimada, A., and Taniguchi, R. (2017). An easy-to-setup 3D phenotyping platform for KOMATSUNA dataset. In The IEEE International Conference on Computer Vision (ICCV) Workshops.

Figure legends

Figure 1. Schematic of the proposed deep architecture. (A) a modality branch, consisting of ResNet50 (He et al., 2016), extracts modality-dependent plant features as a feature vector of 1,024 neurons. (B) The fusion part combines those features to retain the most useful information from each modality. (C) The regression part, relates fused information with leaf count as a non-linear regression. (Best viewed in color online).

Figure 2. Leaf count prediction in the CVPPP dataset (all images altogether). (a) Ground-truth vs. prediction, shown as a scatter plot. Due to integer values color shows how many points are overlapping. Dashed parallel lines show the ±1 leaf error range. Note that our approach has high agreement w.r.t. the real leaf count. (b) error distribution. Observe that there is 83% chance that the error will be ±1 within 0 (highlighted area), a number close to

the agreement among human observers (~90%; Giuffrida et al., 2018). (Best viewed in color online).

Figure 3. Error distribution of our network fine-tuned using Tobacco plants in A3 dataset (Minervini et al., 2016). We reported the distribution of the error committed in the testing set, after refining the network parameters with 7 Tobacco plants (a), 14 (b), 21 (c), and 27 (d) plants. When we train with more images (>=21), the highlighted area (error up to ±1 leaf, *c.f.* Figure 2) contains more than 80% of the cases. (Best viewed in color online).

Tables

Table 1. Testing set results of *PhenoDC* trained on RGB images from the CVPPP 2017 dataset (Minervini et al., 2016; Bell and Dee 2016; Scharr et al., 2014). *DiC* and |*DiC*| report mean and standard deviation (in parenthesis) and lower is better. MSE, lower is better. Percentage Agreement (%), higher is better.

		A1	*A2*	*A3*	*A4*	*A5*	*All*†
DiC ↓	***PhenoDC (this paper)***	**-0.39 (1.17)**	**-0.78 (1.64)**	**0.13 (1.55)**	**0.29 (1.10)**	**0.25 (1.21)**	**0.19 (1.24)**
	Giuffrida et al. 2015	-0.79 (1.54)	-2.44 (2.88)	-0.04 (1.93)	-	-	-
	Romera-Paredes and Torr 2016	0.20 (1.40)	-	-	-	-	-
	Aich and Stavness 2017	-0.33 (1.38)	-0.22 (1.86)	2.71 (4.58)	0.23 (1.44)	0.80 (2.77)	0.73 (2.72)
	PhenoDC (this paper)	**0.88 (0.86)**	1.44 (1.01)	**1.09 (1.10)**	**0.84 (0.76)**	**0.90 (0.85)**	**0.91 (0.86)**
	Giuffrida et al. 2015	1.27 (1.15)	2.44 (2.88)	1.36 (1.37)	-	-	-

|DiC| ↓	*Romera-Paredes and Torr 2016*[‡#]	1.10 (0.90)	-	-	-	-	-
	Ren and Zemel, 2017[‡#]	0.80 (1.10)	-	-	-	-	-
	Aich and Stavness 2017	1.00 (1.00)	**1.56 (0.88)**	3.46 (4.04)	1.08 (0.97)	1.66 (2.36)	1.62 (2.30)
MSE ↓	***PhenoDC (this paper)***	**1.48**	**3.00**	**2.38**	**1.28**	**1.53**	**1.56**
	Giuffrida et al. 2015	2.91	13.33	3.68	-	-	-
	Aich and Stavness 2017	1.97	3.11	28.00	2.11	8.28	7.90
% ↑	***PhenoDC (this paper)***	**33.3**	11.1	**30.4**	**34.5**	**33.2**	**32.9**
	Giuffrida et al. 2015	27.3	**44.4**	19.6	-	-	-
	Aich and Stavness 2017	30.3	11.1	7.1	29.2	23.8	24.0

[†] A paired t-test between our method and *Aich and Stavness 2017* (the only two approaches from the CVPPP Workshop 2017) shows statistically significant differences (p-value <0.0001).

[‡] Trained on A1 only.

[#] Training and inference are performed using per-leaf segmentations and not total leaf count as with the other methods.

↓ best values are those closer to 0.

↑ best values are those closer to 1 (or 100% in the case of *Percentage Agreement*).

Bold results show best performance.

Table 2. Testing performance of PhenoDC on the multi-modal dataset (Cruz et al., 2015). We report results when the network is trained using only a single modality and when also using all the three modalities.

Training on	DiC ↓	|DiC| ↓	MSE ↓	% ↑
RGB only	0.02 (0.75)	0.48 (0.57)	0.56	55.7
FMP only	-0.06 (0.72)	0.45 (0.56)	0.52	58.7
NIR only	0.13 (0.61)	0.33 (0.53)	0.39	69.6
All (RGB, FMP, NIR)	0.11 (0.40)	0.13 (0.39)	0.17	88.5

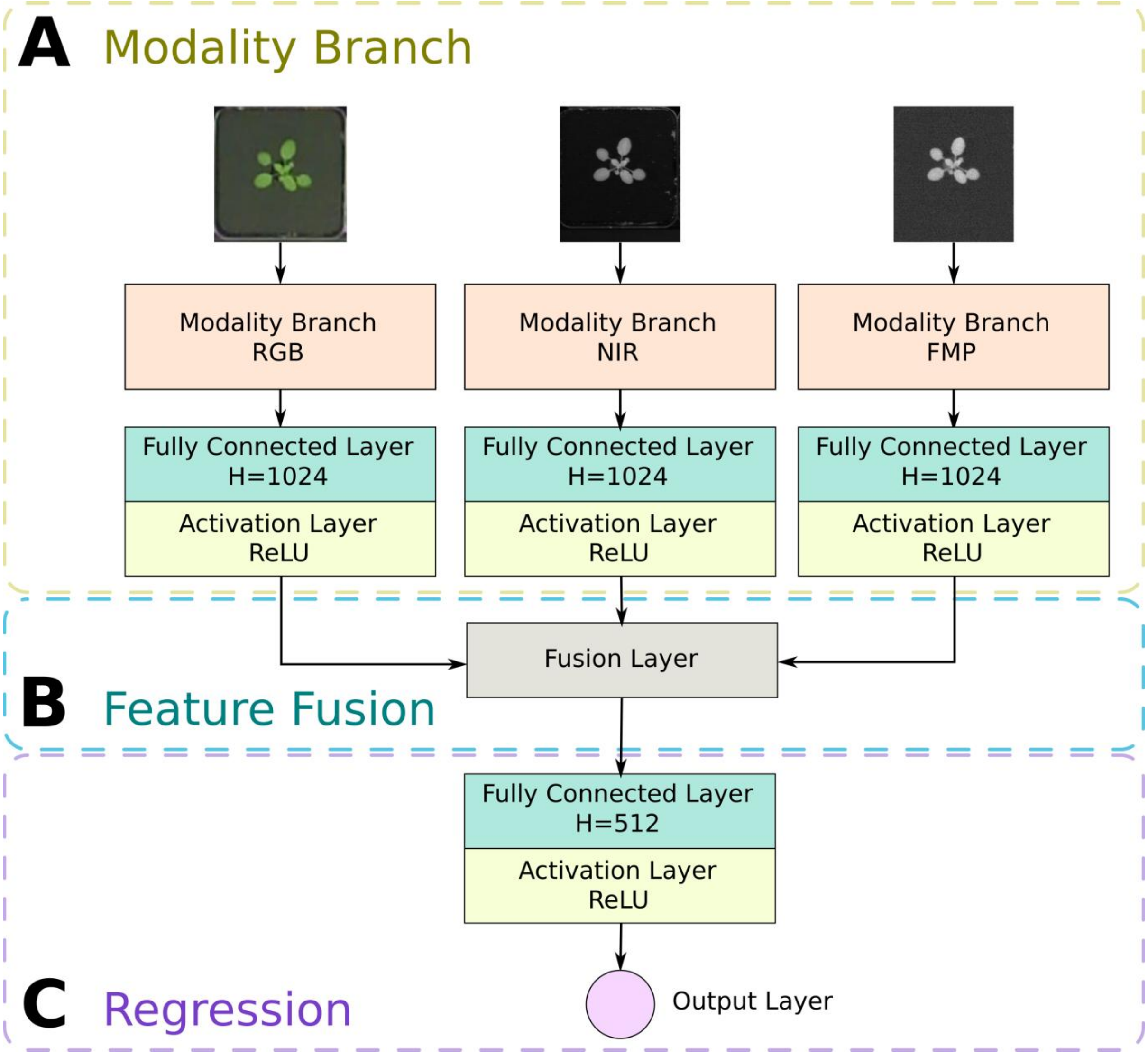
Table 3. Adapting (fine-tuning) the parameters of the proposed architecture to work on Tobacco images (A3 dataset (Minervini et al., 2016)) previously pre-trained with Arabidopsis plants A1, A2, and A4 (Bell and Dee, 2016; Minervini et al., 2016). We progressively increase the number of training images to find a suitable number of images required to create a meaningful model that can count Tobacco leaves. Below we report the results on the held-out testing set.

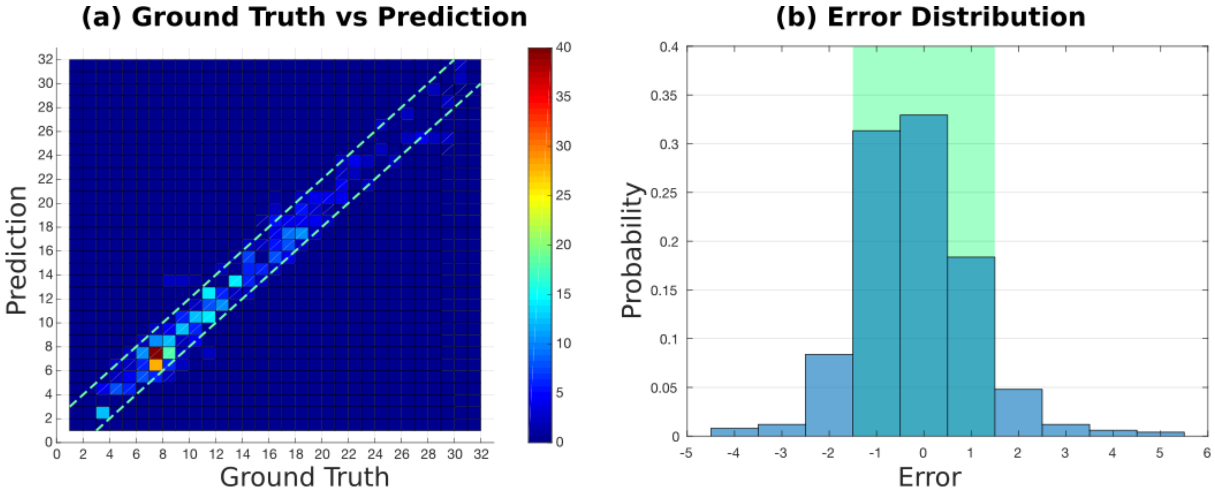
# of training images	DiC ↓	|DiC| ↓	MSE ↓	% ↑
7	-0.39 (1.65)	1.32 (1.07)	2.83	23.2
14	0.00 (1.32)	0.96 (0.90)	1.75	32.1
21	0.27 (1.36)	0.87 (1.07)	1.91	41.1
27	0.25 (1.20)	0.86 (0.87)	1.50	37.5

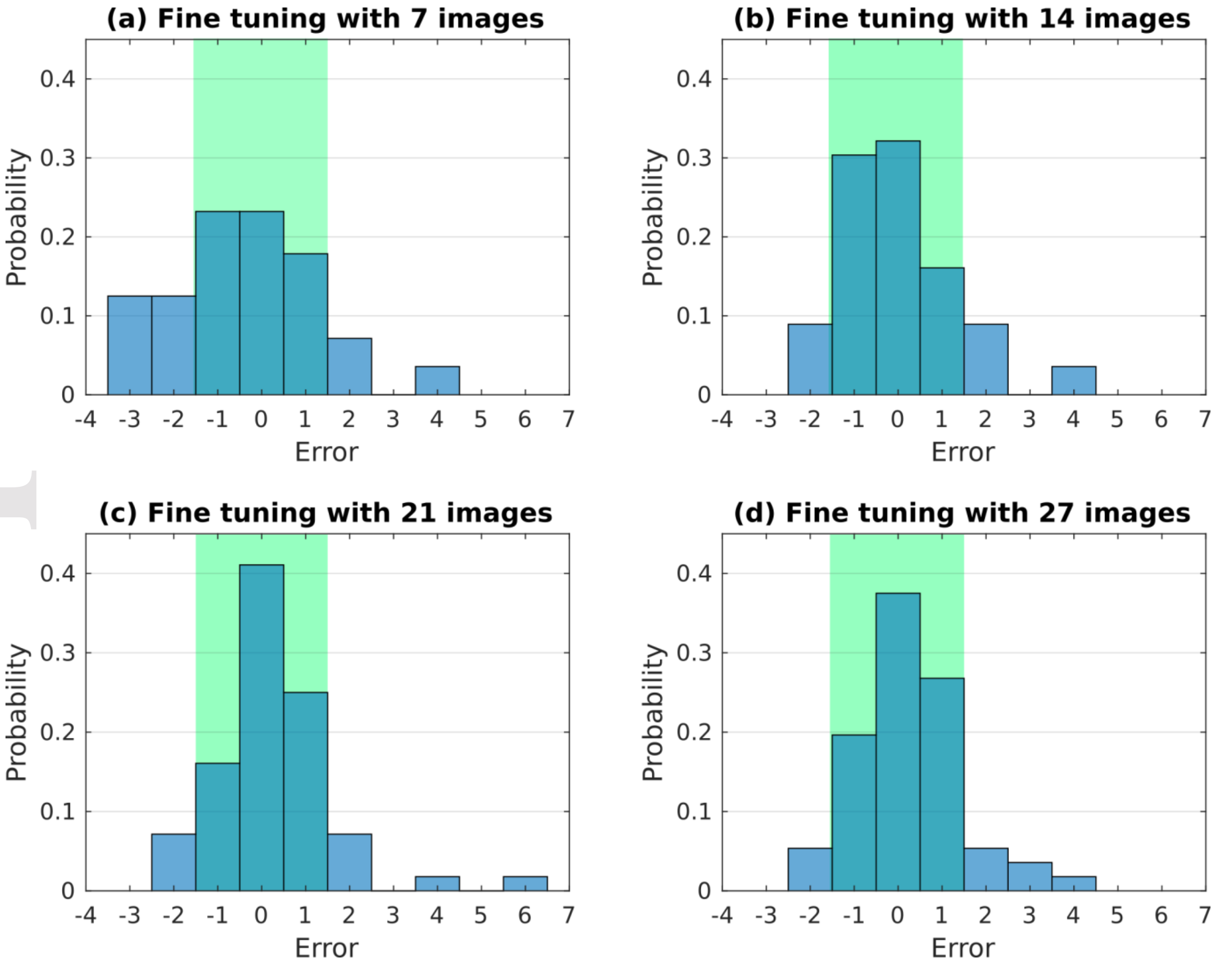
Table 4. Similar process as described in Table 3 but repeated for Komatsuna plant leaf counting based on data available in *Uchiyama et al., 2017.* Model has been trained on Arabidopsis as described in Table 3. Results shown refer to testing set.

# of training images per plant	*Hours of the day*	DiC ↓	|DiC| ↓	MSE ↓	% ↑
10	3pm	-0.74 (1.08)	0.96 (0.89)	1.71	35.0%
20	3pm and 11am*	-0.54 (0.95)	0.86 (0.65)	1.19	25.0%
30	3pm, 3 am*, and 11am*	0.18 (0.92)	0.67 (0.66)	0.88	44.2%
40	3pm, 3 am*, 7am*, and 11am*	0.24 (0.84)	0.59 (0.64)	0.76	49.1%

* Images taken the next day.


A Modality Branch
Modality Branch RGB
Modality Branch NIR
Modality Branch FMP
Fully Connected Layer H=1024
Activation Layer ReLU
Fully Connected Layer H=1024
Activation Layer ReLU
Fully Connected Layer H=1024
Activation Layer ReLU
Fusion Layer
B Feature Fusion
Fully Connected Layer H=512
Activation Layer ReLU
Output Layer
C Regression


(a) Ground Truth vs Prediction
Prediction
Ground Truth
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32
0 5 10 15 20 25 30 35 40
(b) Error Distribution
Probability
Error
0 0.05 0.1 0.15 0.2 0.25 0.3 0.35 0.4
-5 -4 -3 -2 -1 0 1 2 3 4 5 6


(a) Fine tuning with 7 images
(b) Fine tuning with 14 images
(c) Fine tuning with 21 images
(d) Fine tuning with 27 images
Probability
Error
0 0.1 0.2 0.3 0.4
-4 -3 -2 -1 0 1 2 3 4 5 6 7