Digital Phenotyping in Plant Breeding: Evaluating Relative Maturity, Stand Count, and Plant Height in Dry Beans (*Phaseolus vulgaris* L.) via RGB Drone-Based Imagery and Deep Learning Approaches

Leonardo Volpato (leonardo.volpato@corteva.com)
Michigan State University, Department of Plant, Soil, and Microbial Sciences, East Lansing, MI 48824 https://orcid.org/0000-0003-1119-0615
Evan M. Wright
Michigan State University, Department of Plant, Soil, and Microbial Sciences, East Lansing, MI 48824
Francisco E. Gomez
Michigan State University, Department of Plant, Soil, and Microbial Sciences, East Lansing, MI 48824 https://orcid.org/0000-0002-2862-7118

Research Article

Keywords: Phaseolus vulgaris, Time series, Convolutional neural networks, Local regression, Segmented regression, OpenCV, Watershed segmentation, Object detection, Elevation models, Vegetation indices

Posted Date: July 13th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-3160633/v1

Abstract

Background

Significant effort has been made in manually tracking plant maturity and to measure early-stage plant density, and crop height in experimental breeding plots. Agronomic traits such as relative maturity (RM), stand count (SC) and plant height (PH) are essential to cultivar development, production recommendations and management practices. The use of RGB images collected via drones may replace traditional measurements in field trials with improved throughput, accuracy, and reduced cost. Recent advances in deep learning (DL) approaches have enabled the development of automated high-throughput phenotyping (HTP) systems that can quickly and accurately measure target traits using low-cost RGB drones. In this study, a time series of drone images was employed to estimate dry bean relative maturity (RM) using a hybrid model combining Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) for features extraction and capturing the sequential behavior of time series data. The performance of the Faster-RCNN object detection algorithm was also examined for stand count (SC) assessment during the early growth stages of dry beans. Various factors, such as flight frequencies, image resolution, and data augmentation, along with pseudo-labeling techniques, were investigated to enhance the performance and accuracy of DL models. Traditional methods involving pre-processing of images were also compared to the DL models employed in this study. Moreover, plant architecture was analyzed to extract plant height (PH) using digital surface model (DSM) and point cloud (PC) data sources.

Results

The CNN-LSTM model demonstrated high performance in predicting the RM of plots across diverse environments and flight datasets, regardless of image size or flight frequency. The DL model consistently outperformed the pre-processing images approach using traditional analysis (LOESS and SEG models), particularly when comparing errors using mean absolute error (MAE), providing less than two days of error in prediction across all environments. When growing degree days (GDD) data was incorporated into the CNN-LSTM model, the performance improved in certain environments, especially under unfavorable environmental conditions or weather stress. However, in other environments, the CNN-LSTM model performed similarly to or slightly better than the CNN-LSTM + GDD model. Consequently, incorporating GDD may not be necessary unless weather conditions are extreme.

The Faster R-CNN model employed in this study was successful in accurately identifying bean plants at early growth stages, with correlations between the predicted SC and ground truth (GT) measurements of 0.8. The model performed consistently across various flight altitudes, and its accuracy was better compared to traditional segmentation methods using pre-processing images in OpenCV and the watershed algorithm. An appropriate growth stage should be carefully targeted for optimal results, as well as precise boundary box annotations.

On average, the PC data source marginally outperformed the CSM/DSM data to estimating PH, with average correlation results of 0.55 for PC and 0.52 for CSM/DSM. The choice between them may depend on the specific environment and flight conditions, as the PH performance estimation is similar in the analyzed scenarios. However, the ground and vegetation elevation estimates can be optimized by deploying different thresholds and metrics to classify the data and perform the height extraction, respectively.

Conclusions

The results demonstrate that the CNN-LSTM and Faster R-CNN deep learning models outperforms other state-of-the-art techniques to quantify, respectively, RM and SC. The subtraction method proposed for estimating PH in the absence of accurate ground elevation data yielded results comparable to the difference-based method. In addition, open-source software developed to conduct the PH and RM analyses can contribute greatly to the phenotyping community.

1. INTRODUCTION

Dry edible beans (*Phaseolus vulgaris* L.) are a significant legume crop consumed directly by humans worldwide and contribute to addressing food security concerns [1]. The United States plays are major role an exporter, with approximately 30% of its total dry bean production being sent abroad each year [2]. There are ten major market classes of dry beans, with pinto, navy, and black beans accounting for 60% of the total production. In terms or production, Michigan is recognized as the second-largest producer [2,

3] and over 80% of Michigan's dry bean production is rainfed and primarily located in the Saginaw Valley, a region situated in the central-east portion of the state near Lake Huron. Consequently, the area experiences a variable climate zone, which poses unique challenges for bean cultivation.

The inherent variability of dry bean production areas presents a substantial challenge for breeders seeking to improve crop yield [4]. As a short season crop that takes 85 to 100 days to mature on average, dry beans are susceptible to various agronomic and environmental stresses that can negatively impact both dry bean yield and seed quality. Consequently, it is crucial to evaluate dry bean cultivar adaptability across target production environments to identify superior lines. This is particularly relevant since a cultivar's success is contingent upon its ability to thrive under the specific conditions of its intended production environment. Therefore, days to maturity or relative maturity (RM) is a crucial trait to measure in the field during cultivar development.

Cultivar selection also depends on various biotic stress conditions such as disease resistance. In 2020, Michigan dry bean producers considered root rots, caused by a complex of soil-borne pathogens, as the number two disease constraint in dry bean production [5] causing yield loss of up to 84% under environmental conditions favorable for soil-borne pathogens such as the *Fusarium* root rot [6]. Timely estimation of dry bean stand count (SC) at early growth stages provides useful information for important agronomic decision-making analyzing the survival rate after root rot loss due to damping-off, as well as plant population estimation. Thus, tolerant genotypes can be selected under disease pressure by determining the final number of emerging plants through visual inspection.

To reduce the damage caused by biotic stress, plant architecture also can be used to provide natural avoidance for disease infestations, as well as optimize planting density, irrigation, nutrient management, and light interception [7, 8]. Additionally, accurate information on plant height (PH) can help breeders to select and develop new dry bean cultivars with improved plant habits (determinate or indeterminate) and plant growth types (i.e., bush, upright vine, prostrate vine) [9]. This can lead to increased yield, better adaptation to specific environmental conditions, and improved resistance to lodging and other agronomic stresses [10]. Therefore, selecting dry bean cultivars with appropriate PH characteristics, such as upright architecture of plant growth (Type II), can increase not only the natural plant defense but also facilitate efficient harvest operations and reduce seed losses [11].

However, traditional approaches to measure RM, SC and PH at field level are limited to visual scores that are subjective, laborious, time-consuming, and costly. High-throughput phenotyping (HTP) using unmanned aerial system (UAS), or drones, has emerged as an essential tool in plant breeding, providing fast, accurate, and cost-effective measurements of various breeding traits [12, 13]. By automating data collection and analysis, HTP enables breeders to rapidly identify and select superior genotypes, thus accelerating the breeding process and reducing costs associated with manual measurements [14–16].

Traditional methods for estimating RM in agronomic crops using drone-based imagery involve the use of multiple linear regression, vegetation indices and threshold values [17–19]. Recently, deep learning (DL) approaches [20], such as convolutional neural networks (CNN) [21, 22] and long short-term memory networks (LSTM) [23], have been introduced for RM prediction, offering improved accuracy and adaptability over traditional methods [24]. Convolutional Neural Networks has also been used to estimate RM using image classification [25], as well as deep image feature extraction has been combined with shallow image feature information to improve RM predictions [26]. By extracting relevant features from images without relying on predefined vegetation indices, these models are more robust to variations in environmental conditions and genotypes. Nevertheless, DL models to estimate RM have been ignoring the temporal relationship between drone flight dates and environmental conditions (e.g., Grau Degree Days - GDD). An intermediate-level feature fusion based on Deep Neural Network (DNN) [27] can add in parallel stream structure that contains the image and environment conditions (GDD) features subnetworks, in which two modalities are eventually combined into a joint representation by a concatenation layer. Therefore, the CNN-LSTM model could be improved by adding weather information such as GDD under environmental stress.

Deep convolutional neural network models also have been widely applied as a reliable analytical method to detection and counting of target structure in several crops [28–31]. In particular, Faster R-CNN has shown promise as an effective method for detecting plant structures and/or counting individual plants using drone images [32–35]. However, there is a limited number of studies in the literature demonstrating the reliability of DL models for the identification of legume crops such as dry beans and soybeans. Therefore, based on the performance of previous studies, SC can be estimated accurately through high-resolution images across experiment fields under different environmental conditions. Object detection using DL approaches have the potential to outperform

traditional techniques based on vegetation indices, thresholds and image segmentation, such as watershed segmentation [36, 37] and image processing using python libraries (e.g., OpenCV [38]), by offering higher accuracy and resilience to variations in plant appearance and density [28, 39, 40].

Image features also can be explored by taking into account the morphologic plant structure to estimate PH by using Digital Surface Models (DSM) and Point Cloud (PC). Both DSM and PC data sources have been utilized to PH estimation, each offering distinct advantages in terms of data quality and precision [41–44]. Recent studies have demonstrated the potential of utilizing remote sensing data and advanced modeling techniques for accurate PH estimation in various crops, such as maize [42, 45], wheat [45–47], sorghum [48, 49], cotton [50], barley [51], and rice [52]. Drone-based PH estimation has proven to be highly accurate and efficient, enabling breeders to rapidly collect and analyze large-scale PH data [13, 53, 54]. However, the accuracy of height measurements from DSMs can be influenced by inconsistencies in ground altitude [53]. Some methods generally require data from a flight before crop establishment, such as soil elevation or a Digital Surface Model (DSM) to calculate PH using a Crop Surface Model (CSM) [54]. Alternatively, ground points can be identified based on the distribution of the height values within the plot and subtracted on a per plot basis to extract the height of the plot [53]. Therefore, a method to classify DSM and PC into ground and non-ground (vegetation) data was also implemented in this study to improve the PH estimation in the absence of accurate georeferenced corrections such as RTK/PPK (Real-Time Kinematic/Post processed Kinematic) or GCP (Ground Control Points) at plot level.

In plant breeding, the integration of drone imagery and advanced DL models can significantly enhance the efficiency and precision of phenotyping pipelines for detecting objects and tracking senescence. In addition, PH, RM and SC can also be measured with traditional empirical methods which can achieve reliable results and reduce computational requirements. The primary goal of this study is to create a data-driven support system for plant breeders that employs HTP pipelines using RGB images to efficiently estimate RM, SC, and PH. The study also aims to compare conventional methods with DL models for accuracy assessment and develop user-friendly HTP tools for the plant breeding phenotyping community. Additionally, this study aims to contribute to the development of more efficient and accurate phenotyping methods as follows:

1. Explore the optimal flight frequency, image resolution and data augmentation for evaluating RM, balancing data quality with resource efficiency.
2. Leverage data augmentation and pseudo-labeling techniques to enhance DL models performance and accuracy in predicting phenotypic traits.
3. Investigate alternative approaches for estimating PH when initial flight data is unavailable or limited, maximizing the use of available information.
4. Create open-source, app based, user-friendly high-throughput phenotyping tools and pipelines that can be readily implemented and adapted by the broader plant breeding community.

These refined objectives serve to address critical aspects of high-throughput phenotyping, from optimizing data collection methods to enhancing model performance and ensuring the developed tools are accessible and practical for routine use in plant breeding programs. By harnessing the power of HTP and DL technologies, breeders can make more informed decisions and accelerate the development of new crop cultivars with desirable traits.

2. MATERIALS AND METHODS

2.1. Experimental design and ground-based data of relative maturity, stand count, and plant height

A set of Black and Navy market class dry bean trials from both standard yield trials (AYT) and preliminary yield trials (PYT) conducted by the Michigan State University (MSU) dry bean breeding program were measured for relative maturity (RM), stand count (SC), and plant height (PH) for ground-truth notes (GT) and aerial imagery data. Data for RM and PH were collected over three growing seasons (from 2020 to 2022) at two locations in Michigan, USA: Saginaw Valley Research and Extension Center (SVREC) (43° 24' 14.9"N, -83° 41' 49.95"W), and an on-farm trial site in Huron county (43°50'42.4"N, -83°15'18.2"W; and 43°51'52.7"N, -83°18'15.4"W) while the SC experiment was performed only in 2022 at SVREC. Trials consisted of breeding lines and

check cultivars ranging from early (81 days) to late (103 days) and short (24 cm) to tall (68 cm) breeding lines. Additionally, the plant population ranged from 66 to 166 considering the annotations boxes and GT data (Additional file 1: Figs. S1-S3). Entries were planted in four-row plots of 4.5 m in length and 0.5 m between rows in which the two-center rows represent the breeding line and the outside rows the border. Alleys between the plots were perpendicular to the rows and were 2.3 m wide. All experimental trials were laid out in an alpha lattice design except the PYT in 2020, which was carried out in an augmented block design due to limited seed availability. Each entry was replicated four times within AYT and three times within PYT, except the PYT in 2020 had only one replication. Trials received industry-standard seed treatments, fertilization, and weed control applications at recommended rates.

Ground-truth visual RM dates were taken by trained dry bean students and technicians visiting plots and manually scoring plots as they matured, eventually assigning a date of maturity to each plot. Relative maturity GT data were rated as soon as the earliest maturing varieties in each location began to senesce. A dry bean plot was determined as mature when plants had senesced and reached physiological maturity, pods were dried and exhibited their mature color, and seeds reached a moisture content at which harvesting is possible without considerable damage [55]. Plots were evaluated twice per week, and maturity dates were interpolated when it was clear a plot reached maturity in between site visits. The date of plot maturity was expressed as the number of days after July 31 corrected by the planting date for all environments. Ground-truth visual SC were measured manually by counting germinated plants with fully expanded unifoliate leaves at the VC growth stage. Manual SC was carried out in 2022 at SVREC location in replications 1 and 2 of standard yield trials only. The limited scope of the SC ground-truth data set was chosen due to the time-consuming nature of rating this trait manually. Plant height was measured with a meter stick from randomly selected plants representative of the average plant height inside each plot, measuring the distance from the soil surface, avoiding any mounds or cracks in the soil, to the tip of the plant. The Android application Fieldbook was used to record all ground-truth data in the field [56].

2.2. UAS platform and image data collection

The UAS imageries were collected using a DJI Phantom 4 Pro v2 (DJI Technology Co., Ltd.) equipped with digital red-green-blue (RGB) cameras (5472x3648) and sensor dimensions of 12.833 (mm) x 8.556 (mm). Flights were conducted after the germination in VE and VC growth stages and across the same time range as RM ground-truth data were recorded in the field from the beginning of maturation to the complete maturation growth stage. Flights were performed within 1 h of solar noon to limit shadow effects, not exceeding 25 min in total flight duration. In 2020, flights were conducted once per week, while in 2021 and 2022, we aimed to fly two times per week. The exact number of days between flights slightly varied if weather conditions (rain, wind) prevented flying. The flight frequency in 2021 and 2022 allowed the simulation of the RM data analysis using 6 and 9 flight missions in total for each year. Waypoints and flying routes were automatically generated using the flight planning software DJI GSP Pro. An iPad was used to prepare missions prior to going to the field. The missions were uploaded to the main control board on the UAS before each flight. The UAS traveled a predetermined route during each flight via the software's autonomous flying mode. To ensure accurate georeferencing of images, permanent ground control points (GCPs) were placed throughout field corners and centers at regular intervals covering the entire field trial area. The GCPs were surveyed with a Global Navigation Satellite System (GNSS) receiver using a real-time kinematic (RTK) correction (Trimble R4 GNSS system, Trimble, Sunnyvale, CA, United States). Experimental plot data and the flight mission settings ranging throughout the year and location are displayed in Table 1.

Table 1
Environment details and flight settings used in this study to estimate relative maturity (RM), stand count (SC) and plant height (PH) traits. A total of five environments were measured between 2020 and 2022 to estimate RM and PH, while SC was estimated using data from 2022 at SVREC only. Each field site varied the total number of plots, planting date, number of ground control points (GCPs), and flight mission settings. Locations in Saginaw (SVREC), and Huron counties, Michigan.

Site	Plots	Plant. date	Flight dates	GCP	Altitude (m)	Speed (m/s)	Overlap (%)	GSD (cm)
2020 SVREC	584	Jul 5 and 17	Aug 14, Aug 21, Aug 27, Sep 3, Sep 9, Sep 17	9	36	4	85–85	0.98
2021 SVREC	524	2-Jul	Jul 14, Jul 19, Jul 28, Aug 2, Aug 13, Aug 18, Aug 23, Aug 26, Aug 30	10	20	4	75–80	0.6
2021 Huron	84	11-Jul	Jul 19, Aug 23, Aug 31, Sep 3, Sep 9, Sep 16	8	16	4	75–80	0.4
2022 SVREC	636 (132)*	1-Jul	Aug 18, Aug 20, Aug 23, Aug 26, Aug 30, Sep 2, Sep 5, Sep 8, Sep 13, Sep 16	8	25 (7, 6, 10)*	2	85 – 80	0.7 (0.15–0.2)*
2022 Huron	132	8-Jul	Aug 18, Aug 20, Aug 23, Aug 26, Aug 30, Sep 2, Sep 5, Sep 8, Sep 13	8	25	2	85 – 80	0.7

*Dataset and parameters used to the stand count pipeline

2.3. Image data processing and CNN methods

The initial steps to obtain the plot level data were similar for RM, PH, and SC pipelines. The automated pipeline, including photo alignment, matching, and bundle adjustment were performed using the Pix4D Mapper software (v4.7.5; Pix4D, Prilly) to obtain orthomosaics and digital surface model (DSM) (Fig. 1-a). The WGS 84 datum was used with a projected coordinate system according to the location UTM zone. Images were imported into Pix4D and then optimized and matched using pre-established adjusted parameter options for generating georeferenced orthomosaics, DSM and Point Cloud data (RGB Pix4D template at Additional file 2: Data S1). The GCPs were input and identified into the Pix4D project using the basic manual editor before initial processing.

R [57] software integrated with QGIS [58] was used to generate the polygon shapefiles according to plot boundary delimitation using the function *'Draw plots from clicks'* available at https://github.com/diegojgris/draw-plots-qgis (Fig. 1-b). Shapefiles were defined using images collected from the first flight available from each location. GDAL (Geospatial Data Abstraction Library) tool plugin in QGIS was used to spatial polygon vectors (or shapefiles) adjustments with a buffer zone for each plot to prevent any influence of neighboring plots. Additionally, plot boundary shapefiles were manually aligned as needed to fit on the two center rows.

For each flight from the RM pipeline, plot images were cropped from the orthomosaic, and then the data extracted was used to create a time series of plot images. Therefore, each dry bean line is snipped into a single image and then sequenced together by drone flight date. A Python script was developed to efficiently import the orthomosaic, crop the plots using the shapefile, resizing images plot to make the data consistent and save the images plot using the plot ID information and their correspondent number of days after the planting (Fig. 1-c and Additional file 2: Data S2 - a._ClipPlots). Stand count plots were clipped similarly to the RM pipeline but using 640x3872 plot sizes instead of 64x256 or 128x512 as evaluated in RM plots.

The temporal maturity date for each plot across time was extracted using the NumPy library in a stacked array format using five dimensions which contains flight date, plot ID, image height, image width, and RGB bands (Fig. 1-d). Additionally, the ground-truth data was adjusted as NumPy format to use as input in the model (Additional file 2: Data S2 - b._MatDatasetNp). Then, the time distributed convolutional neural networks automatically extract the image features for each plot image and the long short-term memory (LSTM) recurrent neural networks are used to predict the relative maturity of beans lines given features extracted using the Tensorflow in Python (Fig. 1-e).

A single plot was used to annotate all visible beans plants using the VGG image annotator [59] drawing a boundary box to perform the SC pipeline in this study (Fig. 1-f). Then, the annotated dataset was used to perform the object detection analysis using the Faster Region-based Convolutional Neural Network (Faster R-CNN) in Pytorch with ResNet50 as the backbone which was pre-trained on the Microsoft Common Objects in Context (COCO) dataset (Fig. 1-h). This type of model is a region proposal-based object detection method that uses a region proposal network that shares full-image convolutional features with the detection network [60].

Two different approaches were compared in the PH pipeline: heights estimated from DSM and PC data source (Fig. 1-i, Additional file 2: Data S2 - c._PlantHeigh_functions). In both methods, the output is produced by subtracting the vegetation elevation from the soil elevation. However, when bare soil flights (DTM) are neither available nor have a good image georeferenced correction (primarily for the Z-axis, [54, 61, 62]), the lowest heights values from the DSM and PC data distribution within the plot were assigned to represent the ground (soil elevation), while the data at the top values of heights distribution were assigned to vegetation (vegetation height). Therefore, PH was calculated for each individual field plot by using quantile and median strategies extracted from the data heights values (Fig. 2-j).

2.4. Maturity deep learning model

In this study, the proposed deep learning model by Moeinizade et al. [24] was adapted with modifications to improve the results. In summary, the authors developed a hybrid model to extract image deep features using a time distributed CNNs and capture the sequential behavior of time series data with a LSTM recurrent neural network. As a result, this model can combine both deep learning approaches that can process multiple data points and sequences to predict an output. More details about this state-of-the-art deep learning model can be found in the original model description [24]. Additionally, the same loss function was used in this study to improve the model performance to outliers combining the mean square error (MSE) loss and mean absolute error (MAE) loss in a piecewise manner according to Huber loss function [63].

2.4.1. Maturity CNN-LSTM model design

The CNN-LSTM model used in this study consists of four time-distributed convolutional layers as used in the implemented model [24]. However, to reduce the number of parameters, decrease the loss and also to avoid overfitting the model, we adapted the number of filters, and LSTM layers, as well as the hyperparameters of the model were adjusted. A grid search to select the best filter number in each convolutional layer was performed using the WandB [64] central dashboard (Additional file 1: Fig. S4). WandB is a Machine Learning Operations (MLOps) tool that provides experiment tracking, collaboration, model versioning, hyperparameter tuning, and model deployment to streamline ML workflows and build reliable ML systems. The optimum hidden units from the LSTM model were also selected by a search grid parameter tuning (Additional file 1: Fig. S5). In the CNN model, down sampling was performed by max pooling with a stride of 2. The dropout regularization technique function from the Keras python library was implemented to help prevent overfitting using a 0.2 fraction rate of the input units to drop.

According to the designed model, the output of the last max pooling layer is flattened and used as input features for LSTM layers which has 256 units that output a vector. Finally, a dense layer with 1 neuron was applied to estimate the relative maturity of each plot. This architecture has a total of 452.497 trainable parameters which represent a reduction of 47.71% compared to the original CNN-LSTM model implemented at Moeinizade et al. [24]. The network weights are initialized using Xavier initialization [65], and all activation functions are rectified linear units (ReLU), except for the final layer, which employs a linear activation function. The loss function was optimized using Adaptive Moment Estimation (Adam) with a batch size of 16 and a learning rate of 10^{-3} and decay rate of 10^{-3} [66]. The model underwent training for 200 epochs. The detailed structure of the network used in this study, as well as the model properties, is provided in Additional file 1: Fig. S7.

2.4.2. Data processing and hyperparameters

Before moving on to prediction, the images go through a number of pre-processing steps as shown in the pipeline Fig. 1 (c-d). In the RM pipeline all images are resized to a fixed width and length as the original size can vary according to the plot shapefile. According to Moeinizade et al. [24] the image features extracted using the proposed deep learning model are related to color and shape only, and the image size does not affect the predictions. However, there are no results available by the authors regarding the

image resolution. In order to examine the impact of image size and image features two datasets with varying image sizes of 64x256 and 128x512 were compared. However, due to computational limitations, it was not possible to test an image size larger them 128x512. The image sizes were selected to optimize the retrieval of plot image features, such as shape, while maintaining image resolution consistency with the shapefile plot. This ensured that the extracted features accurately reflected the plot data, enabling more precise analysis and interpretation. In addition to the image size comparison, this study also investigated the effect of flight frequency on prediction performance. To this end, the RM prediction pipeline was trained using the flight data sets from 6 and 9 flights for each of the image sizes under investigation. This approach allowed us to assess the optimal image size, as well as frequency of flights for accurate RM prediction using the developed pipeline.

The input data set was composed of image plots merging five environments of which 80% were randomly selected and the rest of images were used as test data set to evaluate performance. Since the number of plots varies across different environments (Table 1), the input and test data set were balanced using the Python library scikit-learn [67] to maintain the same proportion of plot images selected in each environment. The input data was split into train and validation sets randomly, and 10% of the validation set (10% of input data) was used to monitor the training process.

Twenty percent of the total image train data set was also tested using data augmentation techniques, including changing the brightness and contrast and making blur images to test the robustness of the deep learning model. As discussed by Moeinizade et al. [24], it is very important to test the model prediction accuracy due to the variation of images at the time of collection caused by cloudiness and the relative position between the camera and the sun. The brightness and contrast of images were changed adding a constant to each pixel and the blur image was done using Gaussian smoothing to remove noise.

A search grid over different values of the hyperparameters σ, batch size, learning rate, kernel size, and decay rate was applied to choose the optimal values that minimize errors and improves the model's performance. The search grid results were also displayed and saved using the WandB dashboard (Additional file 1: Fig. S5). The σ is the hyperparameter value from the Huber loss function, and the optimum value was set to 0.1 by using another separate search grid (Additional file 1: Fig. S6). The hyperparameter models were trained for 100 epochs to determine the optimal weights and model parameters, utilizing the same strategy for separating the training, testing and validation datasets, as well as the data augmentation method. This hyperparameter optimization process guaranteed a rigorous assessment of the model's performance and enabled the final prediction model to attain the best possible outcomes. The results were then saved and visualized using the WandB MLOps tool (Additional file 1: Fig. S9) dashboard for further analysis and investigation and the python code to perform the analysis can be found at Additional file 2: Data S2 - d._HyperparameterSearch.

The model performance evaluation was measured using the *MAE* and *MSE* metrics applying the optimized hyperparameters selected. The RM deep learning model was implemented in Python using the Keras library with the Tensorflow backend at Google Colab Pro+. Using the grid search optimal parameters selected, the training model took about an hour on an NVIDIA-SMI Tesla T4 GPU to complete the task. A comprehensive Python script to implement the RM pipeline using DL to perform the predictions for each set of flights and image sizes can be replicated, as demonstrated in Additional file 2: Data S2 - e._MaturityModel.

2.4.3. Benchmark

The RM deep learning model was compared against the date to plant maturity pipeline developed by [19] using a local regression model (LOESS) and segmented linear regression model (SEG). The performance of LOESS and SEG methods was evaluated using the entire dataset available for each environment. In summary, the LOESS model is fitted to the RGB color transformation values over time and SEG model fitted multiple linear models to the data for different ranges of the explanatory variable. LOESS combines nonlinear regression with linear least squares regression by fitting linear models to localized subsets of data determined by the nearest neighbor algorithm [68], while the SEG tests for differences between slopes to estimate the break-points between the linear models aiming for the identification of the senescence phase curve [18].

A user-friendly R Shiny application (matuRity app [69]) was developed to implement the LOESS and SEG models aiming to facilitate and speed up the plant maturity extraction (see also Additional file 2: Data S2 - g._Loess_Seg_R). The matuRity app features an enhanced version of the pipeline from Volpato et al. [19] to improve data analysis when processing different sets of flights (e.g., 6 and 9 flights in this study) and minimize errors or missing values. Furthermore, the app also contains parameters

and plot visualization features to support the researcher on data analysis and threshold selection. The app loops through the cropped plots by a shapefile and runs the VIs extractions using a parallel computational function rather than using the entire orthomosaic to perform the VIs extraction. The mean greenness leaf index (GLI) was extracted from the time series of images (6 and 9 flights date) and the RM was estimated using an optimized threshold value of 0.06. To perform the VIs extractions from each breeding plot in the field, an open source Streamlit app in Python was developed [70] and can be accessed online (https://msudrybeanbreeding-vegetation-index--vi-extractions-v0-3-9knpzt.streamlit.app/). VIs app calculator was designed to accelerate and provide accurate results at plot level extraction (mean, median, standard) and parameters to mask the soil (i.e., HUE and HI and threshold value). The VIs outputs can be exported into a csv file. Additionally, according to the user preferences, a GitHub repository with the RM pipeline and a sample data set also is available (https://github.com/msudrybeanbreeding/Workshop_01_VIs) running an R script to perform VIs extractions analysis (Additional file 2: Data S2 - f._VIs_R).

2.5. Stand count deep learning model

The SC pipeline deployed in this study was performed in six steps from the raw images and annotations to the final SC predictions. Figure 2 shows the SC pipeline and the detailed description in each step. This pipeline was designed to be user friendly containing all the codes and step-by-step method. In total, 132 field plots were measured as ground-truth (GT) and annotated, however, an extra 8 plots were added to the final data set to increase the annotations. Therefore, in total 17259 target single plants were obtained during the annotation step. The VGG annotation tool was used to label the annotated plants from plot images while the manual data measured at field level was used as GT.

2.5.1. Stand count Faster R-CNN model design

Faster R-CNN is a classical object detection method based on R-CNN and Fast R-CNN which was proposed by [60]. It combines RPN with the Fast R-CNN module, that makes it possible to acquire feature maps, generate candidate regions, and perform regression and classification operations all fused into one deep neural network. This method has become one of the most popular object detection frameworks with its advantages of high accuracy and robustness [71]. Compared with the Fast R-CNN, a region proposal network (RPN) structure was added in the Faster R-CNN as a representative two-phase model [60, 72]. The advantage of RPN is the idea of sharing of weights and translation-invariance which not only ensures accuracy but also helps in fast end-to-end recognition [73]. Usually, Faster R-CNN can be divided into four parts: a backbone, RPN network, regions of interest (ROI) pooling, and fully connected classification and regression model [29]. In this study, the ResNet50-FPN backbone was used taking the amount of computation available into account, as well as the capacity to resolve the degeneration problem of deep neural networks and generate a high-quality feature map [74]. Additionally, the ResNet-based backbone has enhanced the detection and recognition accuracy on biological targets [30, 75].

2.5.2. Data processing

The first step in this pipeline, as shown in Fig. 2, consists in reading the VGG annotations and to transform the csv file output to a COCO-style format which have x, y, width, and height dimensions (Additional file 2: Data S2 – h.._ReadingAnnot). Due to the plot boundary (shapefile) variation, some image plots were obtained in slightly different scales. Therefore, in the second step, all plot images were rescaled to a consistent size of 3872 × 640 pixels. (Additional file 2: Data S2 – i._Resize_img_annot). The selected image size was chosen to closely resemble the actual plot shapefile size, while also meeting the Faster R-CNN image size requirements to maximize the information contained within the plot images.

A grid search hyperparameters was deployed to select the optimum batch size, learning rate, and decay weights values using the WandB MLOps tool (Additional file 1: Fig. S8) to save and display the results in a dashboard (Fig. 2 – Step 3, Additional file 2: Data S2 - j._HyperTuning). To this end, the plant detected model was developed using PyTorch 1.8.1 in Python 3.7, and the plot images dataset was randomly divided into the 71.5% to training set and 28.5% to validation set. Therefore, the training and validation sets contained 100 images, and the test set contained 40 images. The hyperparameter model was trained for 100 epochs.

To create a valid competition metric, the performance evaluation using the intersection over union (IoU) [76] was used to both hyperparameters tuning and training model steps (Fig. 2). The result of IoU is the ratio of the intersection region of the candidates and the ground truth to the concatenation region. The ratio is compared with the set threshold, and if the ratio is greater than the

set threshold, the candidate box sample is judged to be a positive sample (detected plant); otherwise, it is a negative sample. Additionally, the "*Albumentations"* Python library [77] also was implemented using '*OneOf'* method to perform a robust image augmentation to increase the quality of trained models. The plot images were transformed using the image augmentation technique pixel-level transforms functions: '*HueSaturationValue', 'RandomBrightnessContrast', 'RGBShift', 'RandomGamma', 'Blur', 'GaussNoise', 'ToGray', 'RandomFog', 'RandomSizedBBoxSafeCrop'*, and *'Cutout'* [77].

In the grid search hyperparameters training process, the momentum parameter for stochastic gradient descent (SGD) was set to 0.9, and the default IoU threshold between the detection box and the ground truth were, respectively, 0.5 and 0.7. The pretrained weights on the COCO dataset were utilized in the training processing for faster convergence and better generalization [78]. The model training and testing to search the optimum hyperparameters were conducted on Google Colab Pro + using a Nvidia Tesla V100 Graphics Processing Unit (GPU) with 16GB Video Random Access Memory (VRAM).

Plant recognition using supervised machine learning can be improved by increasing the amount of annotated data to train. Thus, the remaining 124 plots from the standard yield trial in 2022 (reps 3 and 4) were used to perform the pseudo-labeling annotations in order to increase the plant recognition accuracy by improving the amount of annotated data to train. This step was done using the optimized parameters selected from the grid search hyperparameters training process. However, in order to avoid noisy labels due to false-positive detection, the default IoU threshold between the detection box and the ground truth was set to 0.5 in the inference process. As a result, the jointly boxes annotated data set using a total of 264 plots image provided 30903 recognized plants. This labeling annotated data was used to perform the updated training data set using the total (264) dry beans plots (Fig. 2 – Step 5, Additional file 2: Data S2 - k._PseudoLab).

The final plant detected model was developed using PyTorch 1.8.1 in Python 3.7, and the plot images dataset was randomly divided into 4 folders aiming 75% or 198 plot images to training set and 25% or 66 plots image to testing set. The optimized hyperparameters in the final training model were adjusted using the SGD with a batch size of 2 [79] to loss function optimization process with a learning rate of and decay rate of 10^{-3} and same momentum parameter and IoU as used to tuning the hyperparameters. The model was trained for 1000 epochs and the settings and output were displayed and saved using WandB dashboard (Additional file 1: Fig. S10, and Additional file 2: Data S2 -l._TrainModel). Additionally, a cross-validation technique was employed to assess the model performance, and the influence of the number of epochs on prediction accuracy was investigated by increasing the epochs to 2000, which revealed the presence of overfitting (Additional file 2: Data S2 -l._TrainModel). Finally, the inference model was performed to classify two classes of identified objects, which are plants at early growth stage (bean) and the non-plants that represent the soil (background) using a detection threshold of 0.1 (Step 6, Additional file 2: Data S2 - m._InferenceModel).

2.5.3. Benchmark

In this study, the proposed DL object detection model was compared to traditional methods (e.g., handcrafted [80]) using computer vision expert designs techniques to extract the plant features via segmentation methods for finding object contours with the OpenCV library (OpenCV1 and OpenCV2) [38], as well as the watershed segmentation approach (WS) for detecting pixel intensity values corresponding to regions of an object and determining object boundaries, implemented using the scikit-image library [67], both in Python programming. Furthermore, the function *'fieldCount'* from the R package FieldImageR [81] was adjusted to calculate the number of plants per plot in the software R using a minimum size percentage of plant canopy of 0.01 (Additional file 2: Data S2 - p._StandCount_WS). OpenCV1 method developed first convert images to grayscale, then applies a Gaussian blur with a kernel size of 5x5, uses canny edge detection with an aperture size of 3 and L2 gradient, next use the dilation function with a 5x5 kernel for 100 iterations, and finally morphological function closing using a 3x3 kernel before identifying contours to determine stand counts (Additional file 2: Data S2 - n._StandCount_OpenCV1). Similar, OpenCV2 method converted images to grayscale, applied morphological closing with a 3x3 kernel, utilized Gaussian blur with a 5x5 kernel, performed distance transform using L2 distance and a mask size of 5, and thresholded the resulting image with a factor of 0.3 times the maximum distance value, creating a binary image from which contours were identified to quantify stand counts (Additional file 2: Data S2 - o._StandCount_OpenCV2). WS-py method, in contrast, converted images to grayscale, applied Gaussian blur with a 5x5 kernel, computed the Euclidean distance map of the blurred image, identified local maxima with a minimum distance of 40 pixels, performed connected component analysis using a 3x3 structure, and finally applied the 'watershed' algorithm function from scikit-image library with the

computed markers and mask to segment bean plants, obtaining stand count values (Additional file 2: Data S2 - p._StandCount_WS).

2.5.3.1. Image segmentation

The proposed pipeline to perform stand count using the traditional methods (OpenCV and WS) were performed by using a mask obtained from an image semantic segmentation pipeline in PyTorch using a U-Net architecture [82] with a MobilenetV2 as the backbone [83]. In this case, two classes were designed which are soil and vegetations. The loss function was optimized using Adaptive Moment Estimation (Adam) with a batch size of 3, learning rate of 10^{-3} and decay rate of 10^{-4}. The input data set was composed from the annotated images (140) in which 80% were randomly selected and the rest of images were used as test data set to evaluate performance. The input data was split into training and validation sets randomly, and 20% of the validation set (20% of input data) was used to monitor the training process. The image augmentations [77] and the normalization image (using the Normalize function from the Pytorch library) were applied to the images, incorporating the mean and standard deviation values. The model was subsequently trained for 15 epochs.

The Semantic Segmentation approach was first trained using the images segmented via a vegetation index (ExG – ExR) derived from the RGB digital numbers formula ((2*g) - r - b) - ((1.4*r) - g), where *r*, *g* and *b* are the digital numbers from the bands red, blue and green, respectively, extracted from the visible image (raw RGB images). Then, the image binarization was done considering soil as value 0 and vegetation value 1. It is essential to note that stand count using traditional methods can be accomplished with binarized images. However, for AI purposes using DL models applied to HTP, as well as providing a model that does not depend on a mask (in this study, ExG – ExR) and a threshold to separate soil (background) and vegetation (plants) [84], Semantic Segmentation can be employed in the future to deliver high accuracy for pixel-level classification. This approach is demonstrated in this study (Additional file 2: Data S2 - q._SemanticSeg) and has been utilized by other authors [31, 85, 86]. Evaluation metrics using the mean IoU and pixel accuracy were 0.938 and 0.9967, respectively, providing reliable results for distinguishing between soil and plants (Additional file 1: Fig. S11). Consequently, in this study, a DL model was employed to perform pixel-wise segmentation for classifying plot images into soil and vegetation categories.

2.6. Plant Height

The estimatation of PH was performed using two different data sets: Digital Surface Model (DSM) and Point Cloud (PC). In this study the same set of flights used to perform the RM analysis were deployed in the PH pipeline. However, as the GT measurements were not collected at the same day of flights, the latest flights performed around the GT notes were used to perform this analysis. Therefore, all the flights performed during the measurements in the week were used to get the drone PH results. As a result, the final PH estimations were obtained taking the average of two or three estimated PH values across flights.

2.6.1. DSM and Point Cloud Generation

The acquired images were processed using Pix4Dmapper to generate a DSM and a high-density 3D point cloud using structure from motion (SfM) algorithm [87, 88] and Multi-View Stereo (MVS) [89] algorithms to reconstruct the 3D structure of the scene. The point cloud was later meshed via an algorithm based on Delauney triangulation [90, 91] computed on multiple image scales with noise filtering and a "*sharp*" surface smoothing filter. The DTM or bare soil elevation was generated for 2021 trials only from images collected by a single flight prior to the vegetation emergence. For all trials in this study, the DSM or vegetation elevation was acquired through flights conducted near the physiological maturity stage. Besides the Pix4D adjusted parameters used in this study, the DSM and DTM rasters, as well as the point clouds were computed following the parameters workflow recommended by Pix4D for high-resolution RGB imagery [92]. To calculate the PH using drone images, we utilized the Crop Surface Model (CSM) generated from the difference-based method, as described in Volpato et al. [54]. In cases where the DTM was not available, strategies involving DSM or PC were utilized instead. These approaches allowed for the accurate calculation of PH, even in areas where the terrain data was limited or unavailable.

2.6.2. Plant Height Estimation

In 2021 the CSM (i.e., the height of individual plot surfaces) was obtained by subtracting the DTM from the DSM raster. Data was extracted from the regions of interest by overlapping the CSM and the shapefile [53, 54]. However, in 2020 and 2022 flights the PH

were obtained by classifying the DSM and PC into ground and non-ground (vegetation) points using an assigned quantile of the pixel's distribution for a given plot. Therefore, the assigned pixels or point cloud elevation data from the vegetation were subtracted from the soil assigned data for a particular plot. Vegetation data was selected by a single value defined by a quantile while the soil data was chosen by taking the median value from pixels below the threshold given by a quantile (top lower distribution values) aiming to maximize the PH accuracies and to reduce errors. The parameters employed for estimating PH using the DSM and PC approaches varied across the evaluated environments (refer to Additional file 1: Table S1). Parameter adjustments were made prior to comparing the PH with the GT measurements in each environment, based on the observed correlations (results not shown).

In order to facilitate the parameters test, experimental plot visualization, as well as enhance the PH data analysis into a user friendly experience, a R Shiny application was developed and can be accessed at https://msudrybeanbreeding.shinyapps.io/PlantHeightR/ or running in a local machine [93]. The PlantHeightR application was created to support researchers and students to improve and integrate the HTP analysis easily into the phenotyping pipeline.

2.7. Performance Assessment metrics

The accuracy of the drone-based phenotyping estimations to RM, SC and PH were assessed by calculating the coefficient of determination (R^2) and Pearson correlation (r) between the drone-derived and ground truth measurements. Additionally, the performance of the deployed models to measure RM was done using the mean absolute error (MAE) and mean squared error (MSE), while PH and SC were evaluated using MAE and root mean square error (RMSE) metrics, which are defined as follows:

$$MAE = \frac{1}{N}\sum_{1=1}^{N}|y_i^{pred} - y_i^{GT}|$$

$$MSE = \frac{1}{N}\sum_{1=1}^{N}|y_i^{pred} - y_i^{GT}|^2$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{1=1}^{N}|y_i^{pred} - y_i^{GT}|^2}$$

where, y_i^{GT} denotes the ground-truth measurements for the ith plot, y_i^{pred} is the predicted or estimated value, and N is the total number of observations.

Furthermore, a custom algorithm was employed to remove the outlier's data from PH and SC correlations analysis, as well as denoising the point cloud data during the PH analysis in order to reduce the noise and enhance the overall PC quality. This method utilized the interquartile range (IQR) [94] to identify and filter out the outlier points. Specifically, the difference between the 25th and 75th percentiles (Q1 and Q3) was calculated, and a range of acceptable values were established using a defined multiplier (kout = 1.5) of the IQR. Data points falling outside of this range were considered outliers and subsequently removed from the dataset. This approach facilitated the preservation of approximately > 95% to PH and > 98% to SC of the original data to all conditions, except to SC at 6m flight that showed > 86% data maintained.

3. RESULTS

3.1. Relative maturity performance

The DL model used in this study demonstrated high performance in predicting the RM of plots using both 6 and 9 flights datasets across several environments (Table 2). In general, the CNN-LSTM model performed better than the LOESS and SEG models according to the metrics (correlation coefficient, *r*, coefficient of determination, r^2; mean absolute error, *MAE*; and mean squared error, *MSE*) to assess the model accuracy. As expected, the training data results from the CNN-LSTM model showed the highest correlation when compared to the testing data and the performance of the LOESS and SEG methods. This outcome can be attributed to the model's ability to learn and adjust its parameters specifically based on the training data, ultimately resulting in a more accurate fit for a particular dataset.

The error analysis, employing both MAE and MSE metrics, demonstrated a substantial reduction in the CNN-LSTM models compared to the conventional LOESS and SEG methods for all evaluated environments. For the 2020 SVREC dataset, the

correlation between GT and DL drone-based image predictions achieved the highest values for the testing dataset (*r*: 0.98 and r^2: 0.96), outperforming other methods. However, for this specific environment, two planting dates were conducted (Table 1), in which the correlation could have been influenced by the variation between planting dates, masking the true evaluation capability of the predictions by extending the range of the response variable. In the 2021 SVREC dataset, the CNN-LSTM models also showed better performance, with the CNN-LSTM model attaining an *r* of 0.32 and r^2 of 0.10 in the test data using 6 flights, which was higher than the LOESS and SEG models, however, with larger image data (512x128). In the 2021 HURON dataset, the CNN-LSTM model performed well with the 256x64 image size achieving an *r* of 0.82 and r^2 of 0.68 in the test data. Although the LOESS model had a high *r* (0.87) and r^2 (0.76), the CNN-LSTM model showed lower *MAE* and *MSE* values, indicating improved prediction accuracy. For the 2022 SVREC dataset, the 512x128 CNN-LSTM model demonstrated better performance, achieving an *r* of 0.79 and r^2 of 0.62 in the testing data, surpassing the LOESS and SEG models using 6 and 9 flights. However, the results using 6 flights for both image size showed better correlations and lower errors. The 2022 HURON dataset using the CNN-LSTM model for 6 flights showed better results when compared to 9 flights. Nevertheless, the methods LOESS and SEG demonstrated enhanced correlations for 6 and 9 flight sets. However, according to the results in the other environments, the traditional methods presented lower performance in terms of errors compared to the DL models (*MAE* and *MSE* < 2 for CNN-LSTM models, > 17 for LOESS and > 8 for SEG).

Overall, the results demonstrate that the adapted CNN-LSTM models employed in this study were successful in predicting the RM of plots across diverse environments and flight datasets, particularly when comparing errors using MAE and MSE. The performance of the CNN-LSTM models outperformed the LOESS and SEG models, highlighting the potential of DL approaches in practical applications for dry bean maturity prediction.

Table 2

Performance metrics for different models deployed to predict relative maturity (RM) in five environments (SVREC 2020, 2021 and 2022, and HURON in 2021 and 2022). The Deep Learning models (CNN-LSTM) were trained and tested at two image sizes (256x64 and 512x128), and the methods CNN-LSTM, LOESS, and SEG were evaluated using either 6 or 9 flights. The performance metrics measured include *r* (correlation coefficient), r^2 (coefficient of determination), *MAE* (mean absolute error), and *MSE* (mean squared error). The metrics are presented separately for the training and test datasets.

		6 flights						9 flights					
Env.	Metric	CNN-LSTM (256x64)		CNN-LSTM (512x128)		LOESS	SEG	CNN-LSTM (256x64)		CNN-LSTM (512x128)		LOESS	SEG
		Train	Test	Train	Test			Train	Test	Train	Test		
2020 SVREC	r	0.99	0.98	99.00	0.98	0.96	0.96	-	-	-	-	-	-
	r^2	0.97	0.96	0.98	0.96	0.93	0.92	-	-	-	-	-	-
	MAE	0.54	0.76	0.34	0.76	7.23	6.29	-	-	-	-	-	-
	MSE	0.85	1.21	0.47	1.06	63.84	49.57	-	-	-	-	-	-
2021 SVREC	r	0.90	0.24	0.91	0.32	0.16	0.12	0.84	0.28	0.89	0.18	0.20	0.26
	r^2	0.80	0.06	0.82	0.10	0.03	0.02	0.71	0.08	0.80	0.03	0.04	0.07
	MAE	0.88	2.24	0.82	2.14	5.66	4.11	0.91	2.05	0.70	2.16	5.20	5.50
	MSE	1.55	7.27	1.45	7.03	39.79	24.04	1.90	6.88	1.35	7.18	34.60	37.32
2021 HURON	r	0.95	0.82	0.93	0.61	0.87	0.76	-	-	-	-	-	-
	r^2	0.90	0.68	0.87	0.37	0.76	0.58	-	-	-	-	-	-
	MAE	0.40	1.25	0.43	1.88	7.56	2.94	-	-	-	-	-	-
	MSE	0.46	3.00	0.54	5.38	58.69	16.56	-	-	-	-	-	-
2022 SVREC	r	0.94	0.70	0.94	0.79	0.71	0.73	0.87	0.78	0.95	0.79	0.74	0.68
	r^2	0.88	0.49	0.88	0.62	0.51	0.54	0.76	0.61	0.90	0.62	0.55	0.46
	MAE	0.36	0.98	0.32	0.84	5.59	5.48	0.64	0.91	0.37	0.89	4.56	4.46
	MSE	0.44	1.74	0.41	1.28	35.63	34.03	0.82	1.45	0.32	1.35	23.81	24.74
2022 HURON	r	0.90	0.61	0.92	0.80	0.86	0.86	0.83	0.42	0.93	0.49	0.87	0.77
	r^2	0.82	0.38	0.85	0.63	0.74	0.74	0.69	0.18	0.87	0.24	0.75	0.59
	MAE	0.37	0.96	0.40	0.65	4.31	2.62	0.64	1.08	0.34	1.12	4.12	4.38
	MSE	0.56	1.65	0.51	0.96	19.23	8.69	0.97	1.88	0.41	1.84	17.58	21.23

3.1.1. Weather effect in the relative maturity prediction

In this study, we selected the 6 flights dataset with an image size of 256x64 to investigate the performance of CNN-LSTM model enhanced with Growing Degree Days (GDD) data. The decision to use this dataset was based on the balance between the computational cost and model performance, as well as the practicality of data collection in real-world agricultural applications. Moreover, to ensure a fair comparison, the same plots used to create the testing and training datasets were selected for both DL models.

The results show that the CNN-LSTM + GDD model outperformed the CNN-LSTM model in certain environments, particularly in 2021 SVREC, where the Pearson's correlation coefficient (*r*) increased from 0.24 to 0.48 and the r^2 value increased from 0.06 to

0.23 (Fig. 3). Moreover, the *MAE* and *MSE* values decreased, indicating better prediction accuracy. This suggests that incorporating GDD information improved the model's performance, particularly under unfavorable environmental conditions or weather stress.

However, in other environments, such as 2020 SVREC, 2021 HURON, and 2022 HURON, the CNN-LSTM model performed slightly better or similarly to the CNN-LSTM + GDD model. In 2020 SVREC, both models showed high *r* values of 0.98, and similar r^2 values of 0.96 and 0.95. In 2021 HURON, the CNN-LSTM model outperformed the CNN-LSTM + GDD model with a higher *r* value (0.82 vs. 0.67) and r^2 value (0.68 vs. 0.44). In 2022 HURON, both models showed comparable performances with similar *r* and r^2 values. In 2022 SVREC, the CNN-LSTM + GDD model demonstrated a slightly better performance, with a higher *r* value (0.74 vs. 0.70) and r^2 value (0.55 vs. 0.49). However, the *MAE* and *MSE* values were quite similar between the two models, suggesting that the added GDD information did not significantly improve the prediction accuracy in this particular environment (Fig. 3).

3.2. Stand count performance

The results demonstrate that the deployed Faster R-CNN model is capable of accurately identifying bean plants at an early growth stage, with a strong correlation between the predicted SC to GT and annotated boxes measurements. The model performed consistently across different flight altitudes, with relatively low errors when compared to the traditional segmentation methods such as OpenCV and the watershed algorithm. The correlation between annotation boxes and GT measurements (r = 0.8) is similar to the correlation between predicted drone values at 7 meters and GT measurements (r = 0.79). Furthermore, the predicted drone values at various heights (10 meters and 6 meters) show similar or better performance when compared to annotation boxes (r = 0.81 and 0.82 to 10m and 6m, respectively). The strong correlation performance observed at a 10-meter flight altitude may be attributed to the improved annotation boxes obtained using the 7-meter plot image dataset. However, the flight performed at early growth stage showed poor results to identify plants in the field, mainly because the low image quality to precise located a plant in the image due to the lack of the plant features such as leaf shape and green pixels area. Therefore, the results indicate that the object detection model employed in this study is resilient under varying flight conditions and can maintain accuracy across different flight altitudes. However, it is important to carefully choose an appropriate growth stage and ensure the use of high-quality annotations.

Figures 4 and 5 present the results of our study using different flight heights (7 meters, 10 meters, and 6 meters) and various metrics to correlate annotation boxes with stand count predictions made by the Faster R-CNN deep learning model, as well as comparing them to ground-truth measurements. Additionally, a 7-meter flight was performed early in the season to assess an early growth stage in the impact on the predictions to detect plants (Fig. 5). Based on these results, the performance of the SC object detection analysis varied with flight altitude, with the drone measurements at 7 and 10 meters showing similar results when compared to the annotation boxes, however, the predicted results by drones at 7 meters flight showed reduced error as evidenced by the lowest *MAE* (17.76) and *RMSE* (19.60) values among the tested altitudes (Fig. 4). The comparison of predicted drone measurements at 7 meters with the annotation boxes showed strong performance, though not as optimal as the 10-meter flight. In contrast, the early flight at 7 meters exhibited a weaker correlation with the GT and higher *MAE* (55.73) and *RMSE* (56.73) values. The 6-meter flight also demonstrated relatively high correlation and r^2 values (Fig. 5), however, the error analysis (*MAE* = 25.5, and *RMSE* = 26.2) presented lower accuracy when compared to the predicted by drone at 7 and 10 meters.

The correlation (*r*) between the annotation boxes and the GT was 0.8, with an r^2 value of 0.63, an *MAE* of 21.04, and an *RMSE* of 22.77. The predicted drone measurements at 7 meters had a correlation of 0.79 with both the GT and the annotation boxes, with an r^2 value of 0.63 and 0.62, respectively. The *MAE* and *RMSE* for the predicted drone measurements were 39.11 and 39.97 for GT and 17.76 and 19.6 for annotation boxes, respectively (Fig. 4). The lack of correlations between GT measurements and annotations are also evident in the trend lines shown in Fig. 4. Furthermore, the largest discrepancies in absolute values for the predictions generated by the object detection DL model are observed at the early growth stage, as illustrated by the trend lines in Fig. 5.

3.2.1. Comparing stand count methods

The deep learning-based approach, Pred_Faster RCNN, exhibited the strongest correlation with GT stand count values (r = 0.79) and annotated boxes (r = 0.79). The method also demonstrated the highest coefficient of determination (r^2) for both GT (R2 = 0.63) and

annotated boxes (R2 = 0.62). In terms of error metrics, Pred_Faster RCNN yielded the highest *MAE* (39.11) and *RMSE* (39.97) for GT among all the methods, however, the Pred_Faster RCNN showed better results compared to OpenCV2 and WS_r when compared to the annotation boxes (Table 3).

The WS_py method followed Pred_Faster RCNN in terms of performance, with *r* values of 0.69 and 0.71 for GT and annotated boxes, respectively. The r^2 results for WS_py were 0.47 and 0.51 for GT and annotated boxes, respectively. Also, the method also yielded relatively the lowest error values among the tested methods, with *MAE* of 14.87 for GT and 9.97 for annotated boxes, and *RMSE* of 17.46 for GT and 12.2 for annotated boxes.

Traditional computer vision methods OpenCV1 and OpenCV2 demonstrated lower performance compared to the DL and Watershed segmentation methods. However, the Watershed method in R (WS_r) presented lower accuracy compared to the Watershed segmentation methods in python (WS_py). In summary, the Pred_Faster RCNN deep learning-based approach outperformed the other methods in terms of correlation and accuracy, followed by the conventional WS_py method. Traditional computer vision methods, OpenCV1 and OpenCV2, exhibited lower performance compared to the deep learning and Watershed segmentation methods (Table 3).

Table 3

Performance comparison of various stand count (SC) estimation methods in the context of high-throughput phenotyping. The methods evaluated include a deep learning-based approach (Pred_Faster RCNN), two traditional computer vision techniques using OpenCV (OpenCV1 and OpenCV2), and Watershed segmentation algorithms implemented in Python (WS_py) and R (WS_r). Performance is assessed using four metrics: Pearson correlation coefficient (r), coefficient of determination (r^2), mean absolute error (*MAE*), and root mean square error (*RMSE*). The values are reported for both ground-truth (GT) stand count and annotated bounding boxes.

Method	Metrics	Ground-Truth	Annot. Boxes
		Score	
Pred_Faster RCNN	r	**0.79**	**0.79**
	r^2	**0.63**	**0.62**
	MAE	39.11	17.76
	RMSE	39.97	19.60
OpenCV1	r	0.42	0.49
	r^2	0.18	0.24
	MAE	34.56	15.51
	RMSE	37.38	19.06
OpenCV2	r	0.37	0.42
	r^2	0.14	0.18
	MAE	23.43	34.47
	RMSE	28.71	41.75
WS_py	r	0.69	0.71
	r^2	0.47	0.51
	MAE	**14.87**	**9.97**
	RMSE	**17.46**	**12.20**
WS_r	r	0.48	0.54
	r^2	0.23	0.29
	MAE	28.43	20.51
	RMSE	35.13	25.44

3.3. Plant Height performance

The performance from the two data sources approaches, CSM/DSM and Point Cloud, for estimating PH was evaluated using data from five different environments. The results are summarized in Table 4, where the performance of each data source is assessed using *r*, r^2, *MAE* and *RMSE* metrics. Across the environments assessed in this study, the PC data source demonstrated a marginally better performance compared to the CSM/DSM data. The average correlation results across environments were 0.55 for PC and 0.52 for CSM/DSM. The ranges were also similar, with 0.39 to 0.79 for PC and 0.31 to 0.8 for CSM/DSM across different environments.

The average r^2 values for PC and CSM/DSM were 0.36 and 0.33, respectively, indicating a moderate relationship between predicted and observed PH values in both cases. The average *MAE* values across the five environments were 11.42 for PC and 12.61 for CSM/DSM, while the average *RMSE* values were 12.56 and 13.64, respectively (Table 4). Analyzing by individual environment, the best performance was obtained in 2021 HURON (*r*:0.8 and r^2:0.64) using CSM data and the environment in 2020 SVREC showed the poorest results (*r*:0.8 and r^2:0.64) using the DSM data source.

When comparing the results for estimating PH in 2021 locations using the CSM (derived from DSM - DTM) against the PH solely obtained from the DSM in 2020 and 2022 locations, the correlations between ground-truth measurements and drone-based PH methods demonstrated better outcomes using the DTM for both CSM and PC data sources. The correlations were 0.71 in 2021 locations versus 0.4 in 2020 and 2022 locations for DSM/CSM, and 0.71 in 2021 locations versus 0.45 in 2020 and 2022 locations for PC. However, lower errors for *MAE* and *RMSE* were observed in the implemented method that classified the PH distribution into soil and vegetation (DSM only), with an *MAE* of 14.6 in 2021 locations compared to 11.31 in 2020 and 2022 locations for DSM/CSM, and 13.4 in 2021 locations compared to 10.1 in 2020 and 2022 locations for PC. Similarly, the *RMSE* was 15.1 in 2021 locations compared to 12.6 in 2020 and 2022 locations for DSM/CSM, and 14.1 in 2021 locations compared to 11.5 in 2020 and 2022 locations for PC (Table 4).

These results indicate that the PC data source generally showed better accuracy and performance in most cases, although the differences between the two data sources were relatively small. The choice between the two data sources may depend on the specific environment and flight conditions, as their performance is relatively close in the analyzed scenarios. The investigation of the PH results can also be observed in more detail in the Additional file 1: Figs. S12.

Table 4
Comparison of plant height estimation performance using CSM/DSM and Point Cloud data sources across five environments. The performance is evaluated using four metrics: Pearson correlation coefficient (*r*), coefficient of determination (r^2), mean absolute error (*MAE*), and root mean square error (*RMSE*).

Env.	Metrics	CSM/DSM	Point Cloud
2020 SVREC	r	0.31	**0.39**
	r^2	0.10	**0.15**
	MAE	**8.14**	12.93
	RMSE	**10.11**	15.12
2021 SVREC	r	0.61	**0.63**
	r^2	0.38	**0.39**
	MAE	20.01	**19.15**
	RMSE	20.61	**19.98**
2021 HURON	r	**0.80**	0.79
	r^2	**0.64**	0.62
	MAE	9.14	**7.64**
	RMSE	9.68	**8.29**
2022 SVREC	r	0.44	**0.48**
	r^2	0.19	**0.23**
	MAE	19.40	**14.32**
	RMSE	20.37	**15.57**
2022 HURON	r	0.47	**0.48**
	r^2	0.23	**0.23**
	MAE	6.38	**3.06**
	RMSE	7.44	**3.83**

4. DISCUSSION

The available open source HTP tools, matuRity [69], PlantHeightR [93], and Vegetation index calculator provided in this study, have the potential to facilitate and increase the data analysis performance in plant breeding and related areas. The user can either access them on-line or download the repository at https://github.com/msudrybeanbreeding?tab=repositories. Additionally, the step-by-step pipelines deployed in this study using DL methods are available at the GitHub repositories, as well as the complete data set used to perform the analysis including orthomosaics, shapefiles, ground notes, clipped plots, and programming codes. Thus, researchers may be able to replicate the present method to predict RM and SC by following the description in this paper along with the scripts developed in Python or R software for each stage from these pipelines. Lastly, the vegetation indices calculator tool in Python will considerably accelerate the process to obtain the VIs by using as input the orthomosaic or reflectance maps and the field shapefiles. The app also has the option to remove the soil based on VIs and threshold. The csv output from the app VIs calculator can be promptly used as input data to run the RM analysis at matuRity app.

4.1. Relative maturity

4.1.1. Impact of GDD, Image Size and Flight Frequency

The ability to accurately identify the maturity date of crops is crucial for commercial plant breeding organizations as it facilitates decision-making and agriculture management. Misclassification of dry bean relative maturity can have adverse effects on various aspects, such as harvest operations, seed germination, breeding advancement decisions, and overall genetic gain. We have demonstrated that CNN-LSTM provides a robust approach to estimating dry bean maturity date within ± 1 day off at 3 locations (2020 SVREC, 2022 SVREC, and 2022 HURON) and ± 2 days off at 2 locations (2021 SVREC and 2021 HURON) compared to the actual maturity date in the field without the ambiguity of the LOESS or SEG models due to the setting of arbitrary thresholds and transformation which lead to information loss (Table 2). The CNN-LSTM results demonstrated comparable performance in predicting soybean RM (± 2 days of the GT) [24]. To predict dry bean RM using a time series of UAS images, color serves as the most crucial feature, as dry beans mature when their pods turn brownish. Given that color is a simple feature that can be readily detected through the initial layers of CNN, the DL approach employed in this study successfully delivered reliable performance using a straightforward model architecture. Therefore, the proposed study demonstrates that a cost-effective RGB drone implementation can be employed to predict RM without human subjectivity, utilizing image feature extraction through CNNs.

The incorporation of GDD information into the CNN-LSTM model was intended to improve the prediction of RM by accounting for the effect of temperature on drying down the bean trials. The shallow effect has also been used to improve the DL model performance of UAS-based soybean maturity information monitoring [26]. The environmental conditions represent a significant bottleneck in phenotyping [95]. Therefore, there is an increasing need to explore more robust DL models to handle several effects other than image features and also integrates genotyping information [96]. The CNN-LSTM approach serves as a valuable tool for handling feature extraction from both phenotype and genotype input data, providing essential insights into plant characteristics and genetics [97, 98]. In this study, the results demonstrated improvements in model performance under specific environmental conditions, particularly when dealing with unfavorable conditions or weather stress (e.g., 2021 SVREC), while in others (e.g., 2020 SVREC), the performance remained similar or decreased. Thus, the model performance demonstrated a potential approach to combine DL image features with GDD-based information to better recognize the environment (Fig. 3). However, the effectiveness of adding GDD information varied across different environments and years. The largest benefit from integrating GDD information was noted in the 2021 SVREC environment, under abnormally hot and dry conditions over a prolonged period of time that hastened maturity (mean values of 10–14 days earlier than typical maturity) far beyond typical dry down conditions. This suggests the need for further research and model refinements to optimize the incorporation of GDD and other environmental factors for more consistent performance improvements across various conditions.

Larger image sizes generally provide more detailed information for the model to learn from, which can potentially improve prediction accuracy [99–101]. However, larger images also require more memory and computational power, which may increase training time and the risk of overfitting [95, 101]. Comparing the performance of the CNN-LSTM model using image sizes of 256x64 and 512x128 pixels, we observed varying levels of accuracy across the different environments and flights frequency (Table 2). In general, the 512x128 image size yielded better performance than the 256x64 image size, indicating that higher resolution images can provide more detailed information for the model to extract relevant features. This improvement in accuracy is particularly noticeable in the 2022 Huron environment when six flights were used (Table 2). However, the difference in performance was not always substantial, suggesting that, in some cases, the smaller image size might provide a good trade-off between computational cost and prediction performance.

Higher flight frequency provides more temporal information about the plants' growth, allowing the models to better capture the dynamics of plant maturity. However, increasing the flight frequency can also increase the cost and complexity of data collection. Our study compared the performance of the CNN-LSTM model using six and nine flights to predict RM. As seen in Table 2, the model's performance varies across environments and image size. In general, increasing the flight frequency from six to nine flights improved the model's performance in some environments (e.g., 2022 SVREC at 256x64), while the performance remained similar or decreased in other conditions. This suggests that the additional temporal information provided by the increased flight frequency contributed to the improved prediction accuracy in some environments. This finding suggests that an optimal flight frequency could be environment-specific and might depend on factors such as land topography, weather, and soil conditions [24]. However, the gains in performance should be weighed against the additional costs and logistical challenges associated with increasing

flight frequency. Therefore, it is essential to find an optimal balance between flight frequency and model performance for practical applications in the plant breeding HTP pipelines.

4.2. Stand Count

4.2.1. Plant detection accuracy

Plant recognition using supervised machine learning can benefit significantly from an increased volume of annotated data for training [102–104]. However, obtaining high-quality, consistent annotations is a time-consuming and costly process, often requiring a well-defined multi-stage process involving multiple team members with specialized roles [105]. To expedite the manual annotation process, alternative or supplementary methods, such as transfer learning, weak supervision, or semi-supervised learning (SSL), can be employed [106, 107]. In this study, we utilized pseudo-labeling annotations, a popular SSL approach, to enhance the amount of labeled data [108, 109]. Pseudo-labeling offers a more efficient means to generate additional labeled data from a fully-supervised model, enabling the Faster R-CNN model to benefit from the increased data without incurring the time and resources typically associated with manual annotation [110–112]. By employing pseudo-labeling in our study, we sought to improve the performance of the Faster R-CNN model for SC analysis while maintaining efficiency and minimizing the manual annotation burden.

The reliable performance for the SC pipeline depends on choosing the appropriate growth stage, as well as training dataset quality. In this study, the 7-meter plot images were utilized to train the Faster R-CNN model, demonstrating comparable performance in predicting SC not only at 7 meters but also at 10 meters flight altitude. Low image resolution images hinder the visual plant identification and label. However, the training could be done using the high-resolution and low-resolution image dataset to increase the model predictive performance. Future directions to increase the accuracy of the model can be adopt training dataset increase techniques such as synthetic high-resolution domain using bicubic interpolation or super resolution techniques [33]. Since Faster-RCNN tends to identify objects with similar sizes to the training datasets, it is important to maintain consistency in resolution and plant size between the training and application datasets. Therefore, under limitations to manually perform annotations, high-resolution images must be used in order to improve the plant features identification, as well as data augmentation techniques to deal with images having different resolutions [33].

The results obtained from the 7m flight dataset using the Faster R-CNN model demonstrate its efficiency and accuracy in identifying bean plants at an early growth stage, with a Precision of 0.9374, a Recall of 0.8019, an F1 Score of 0.8641, and Accuracy (*Ac*) of 0.76. With a True Positive (TP) rate of 76.1%, the model effectively detects the presence of plants, while maintaining a low False Positive (FP) rate of 5.08% (Additional file 1: Figs. S13). This low rate of FP indicates that the model seldom misidentifies non-plant objects as plants [113, 114]. Additionally, the False Negative (FN) rate of 18.8% demonstrates that the model misses a relatively small proportion of actual plants in the field. These results provide similar performance as those reported by Velumani et al. [33] who obtained an accuracy of 0.88 and Precision of 0.95, and also to David et al. [80] with $Ac \approx 0.8$. In this study, we employed a GSD of 0.15–0.2 cm, whereas the cited authors used images with a spatial resolution of approximately 0.3 cm. Karami et al. [115] also obtained good results with *Ac* of 0.82 with a spatial resolution of around 1 cm. However, it is important to note that these results were obtained from maize plants, which typically have a lower plant population (due to wider spacing between plants in the field) compared to crops like dry beans or soybeans. Therefore, the results obtained in this study indicate the effectiveness of the model in accurately predicting the presence of plants in the field, as well as its ability to identify a high proportion of the actual plants present. Moreover, the F1 Score and *Ac* confirm its robustness and potential for high-throughput phenotyping applications, enabling accurate SC predictions using low-cost RGB drone imagery.

False negative plant identification greatly impacted the accuracy of the model, accounting for 18.8% of the total errors in predictions (Additional file 1: Figs. S13). Previous studies have emphasized the significant influence of training image resolution on plant identification [33, 116–118]. Lower-resolution images display reduced textural information and green fraction per plant, which can hinder the recognition of plant shape or leaf contours due to high variation in leaf appearances, as well as impede plant separation due to leaf obstruction. To address issues related to image resolution when performing stand count in crops with high-density populations (e.g., dry beans and soybeans), future research could explore the use of super-resolution techniques [119–121]. These techniques aim to enhance image resolution by adding textural information while maintaining image quality and fidelity [122]. Consequently, employing these methods could mitigate the GSD effect, improve the SC pipeline, and reduce field

expenses and resources associated with conducting flights and processing raw images, since fewer images would be required when flying at higher altitudes. A comprehensive error analysis was performed to investigate the overall results, the influence of annotation box size and border annotation boxes, as well as the impact of noisy targets. This analysis can be found in Additional file 2: Data S2 - r._ErrorAnalysis.

Leaf obstruction led to a higher rate of plant misidentification (FN error) in drone images. Canopy images may limit plant labeling due to occlusion by leaves, resulting in some plot images with poor plant annotations that lack corresponding lateral texture features, making it difficult for the model to distinguish plants. This situation was also observed even during field-level measurements (GT measurements), as human subjectivity could lead to inaccurate stand count data. Consequently, the model was better at identifying plants that were easily labeled and minimally affected by external noise. High plant population density can exacerbate obstruction issues, leading to plant annotations based on the canopy surface. As discussed by Li et al. [29], the inconsistent sizes of rectangular labeling boxes and labeling difficulties could also decrease model performance. The presence of weeds, young leaves, and complex leaf scenes resulting in appearance distortion could lead to detection failures and increased FP ratings [30]. However, under the field conditions examined in this study, our model was not affected by plants other than beans, as the fields exhibited good agronomic practices to control weeds (See Fig. 6 in the next session). This issue could be addressed by including more object classes in the model to identify plants, soil, and weeds. To achieve this goal, robust data annotations and training images should be used, though it will demand extra time and effort.

The IoU method measures the overlap between the predicted plant bounding box and the GT plant bounding box. In this study, values higher than the 0.5 IoU threshold indicate that the predicted plant bounding box is considered as TP while values lower than the threshold are FP. Lower values to IoU indicate less accurate plant localization in determining the precise dimensions of dry bean plants, however, an inaccurate estimation of plant dimensions is not critical for those applications assessing emergence rates and uniformity or plant population, where plant density is the targeted phenotypic trait as performed in this study. Consequently, to enhance the performance of the Faster R-CNN model, testing lower IoU threshold values could be beneficial, as demonstrated by Velumani et al. [33], who utilized an IoU threshold of 0.25 to estimate plant density at corn plants. However, the authors also reported that the accuracy to estimate the plant dimension can also be affected by the difficulty of separating the green from the ground in the shadowed parts of the images. When the goal is to evaluate plant size during early developmental stages as well, mask-based RCNN models can be considered as an alternative [123, 124]. Unlike algorithms such as Faster-RCNN, which are trained on rectangular regions, mask-based methods can better handle the shadows projected on the ground. However, creating mask annotations can be time-consuming, leading to increased efforts in generating a diverse training dataset.

The traditional methods (OpenCV and WS) employ a rational combination of image processing techniques, such as edge detection, morphological operations, distance transforms, and connected component analysis. The Watershed algorithm, specifically, involves peak detection, while the other methods focus on contour identification [125]. Through the logical integration of these approaches, bean plants can be segmented, to enable the determination of SC values. The DL model deployed in this study (Pred_Faster RCNN) achieved an r value of 0.79 when compared to GT measurements, while other methods, such as OpenCV1, OpenCV2, WS_py, and WS_r, achieved r values ranging from 0.37 to 0.69 (Table 3).

Traditional methods, such as OpenCV and Watershed (WS), utilize a strategic combination of image processing techniques like edge detection, morphological operations, distance transforms, and connected component analysis. Specifically, the Watershed algorithm focuses on peak detection, while other methods target contour identification [125]. By logical integrating of these techniques, bean plants can be segmented to facilitate SC value determination. In this study, the deep learning model employed (Pred_Faster RCNN) achieved a correlation coefficient (r) of 0.79 when compared to GT measurements. In contrast, other methods like OpenCV1, OpenCV2, WS_py, and WS_r resulted in correlation coefficients ranging from 0.37 to 0.69 (Table 3). Thus, the object detection model exhibited superior performance compared to handcrafted methods for performing SC in crops of high-density population, such as dry beans. However, traditional methods employing Watershed and image processing with OpenCV show significant potential for achieving satisfactory results when analyzing images from spaced crops, such as maize plants. These conventional analyses can be quite appealing due to their ability to provide accurate results while being less computationally intensive to process.

The discrepancy in the absolute values of the evaluated correlations between GT measurements and annotation boxes highlights the limited association between them, likely due to low-quality measurements in either the field or the annotations. Consequently, lower correlations are anticipated in the predicted analysis (Fig. 4 – trend lines). Compared to the image processing methods used in this study (OpenCV and WS), the annotation quality is an important issue when performing object detection analysis [105]. This observation underscores the importance of accurate data collection and annotation to ensure reliable and robust results in object detection studies.

4.2.2. Plant distribution assessment

In addition to stand count estimation, which can help breeders select resistant varieties based on survival rates, this pipeline can also be utilized to gather information on plant locations within a plot to examine spatial plant distribution (PD). Consequently, it enables the calculation of distances between plants, providing valuable insights for optimizing crop growth and management. Assessment of PD can be useful in various applications, including its potential impact on crop yield and overall field management [126]. Uneven distribution in crop population can result in local crowding or insufficient seedlings in the field, leading to a significant decrease in yield [127–129], as well as create disparities in the light environment within the crop population, impacting crop development [130–132]. Maintaining optimal plant spacing can lead to more efficient resource utilization, such as water and nutrients, and contribute to increased grain yield [133]. Furthermore, optimizing PD can minimize competition between plants, promoting uniform growth and development resulting in increasing crop yields [134]. In addition to its potential impact on yield, the SC pipeline's ability to analyze PD can also be used to improve crop models.

Through the application of PD analysis, gaps between plants can be identified, as well as instances where plants are in closer proximity (double or triple plants). Gaining insight into PD patterns can support researchers and breeders in making more informed decisions when selecting traits that contribute to enhanced grain yield and overall crop performance. By incorporating PD data into the prediction model (e.g., use PD as covariate), researchers can produce more precise yield predictions, which in turn can support breeders and farmers in making better-informed decisions related to crop management and cultivar selection. This ultimately results in enhanced agricultural outcomes and overall productivity. Therefore, plot images and boxes annotations used to perform the SC analysis also were deployed to develop the PD analysis (Additional file 2: Data S2 - s._SpatialDist). As a result, a csv format table can be obtained, as well as a plot image visualization containing the plant distances (Fig. 6).

4.3. Plant Height

4.3.1. Plant Height accuracy

The findings in this study align with previous research indicating that the performance of CSM/DSM and PC for PH estimation may vary depending on the specific environment and flight conditions [53, 54, 135]. Overall, the PC data source showed better accuracy in most cases, suggesting that it may be more suitable for PH estimation in HTP applications. However, it is important to consider that the performance differences between the two data sources were relatively close, implying that the choice between them may ultimately depend on factors such as data availability, processing requirements, and environmental conditions.

Parameters related to plant morphology and structure, such as PH, can be derived from 3D reconstructions dense point clouds, orthomosaics, and DSMs, otherwise known as digital elevation models [53]. These methods utilize structure from motion (SfM) algorithms in downstream image analysis, where multiple 2D images are overlapped to generate a 3D reconstruction [136]. The accuracy of PH metrics extracted from SfM PC data is generally superior compared to other approaches [135], a finding that is supported by the results of this study. The estimated PH obtained from the PC dataset showed better performance with higher image resolution (lower values of GSD), as observed in the 2021 experimental trials (Table 1). This is particularly evident in the 2021 HURON trial, which achieved the best correlation between GT and drone-estimated PH across all environments for both DSM and PC methods (r = 0.8, Table 4). However, PC data consists of thousands of data points, which can exponentially increase with higher image resolution and overlap, thus demanding more computational resources. This time-consuming analysis becomes evident when users choose to use PC data instead of DSM for PH analysis with the PlantHeightR app. To expedite the process, users can crop the precise field area where PH is being collected (see the "*Snipping tool*" tab in the app). Nonetheless, as shown in

this study, the DSM dataset can be utilized as an alternative to extracting PH instead of using larger PC data, effectively reducing computational demands.

A high-quality DTM or ground altitude is essential for enhancing PH estimations, particularly when the soil elevation contains inconsistencies. There are several methods to determine ground altitude such as DTM interpolation method, difference-based method, manual measurement self-calibration method, and exposed alley subtraction method [53]. The findings of this study demonstrated that the subtraction method [45] – which assigns height values within a plot and subtracts them on a per-plot basis to extract the PH – provided reliable accuracy for PH estimation using DSM and PC data source. Nevertheless, to decrease absolute error values, this method depends on the availability of unobstructed areas with exposed soil, which may not always be present in all crops and field layouts. Furthermore, this approach allows for the calculation of ground height for each individual plot, reducing the computational intensity needed for interpolating a DSM across the entire field, as implemented in the PlantHeightR app analysis.

Precision positioning information plays an important role in the accuracy and quality of a DSM [137]. As a result, georeferenced correction is highly recommended for improving PH results, particularly when dealing with UAS systems that have less accurate internal GPS. To compensate for UAS GPS inaccuracies, GCPs with high-quality GPS information can be placed in the field. Acquiring accurate soil or vegetation elevation data can be challenging and resource-intensive, particularly in the absence of georeferenced corrections such as RTK/PPK or GCP at the plot level. This approach can be time-consuming, both at the field level and during data processing. The deployed subtraction method in this study showed a good alternative to estimate PH at plot level rather than use the DTM from the difference-based approach. Hence, this method can be employed even in the absence of GCPs, as the analysis relies on the in-situ plot PH value distributions. This flexibility allows for more accessible and efficient estimations without compromising the overall accuracy of the results.

The accuracy of PH measurements using drone imagery is also dependent on which pixels or data are chosen to represent either the top of plants or the soil level altitude. Thus, the erroneous pixels can be removed by selecting the optimal SfM metrics for both DSM and PC plant height extraction. This process can be done by testing several thresholds and extraction methods for soil and vegetation classification. The optimal metric selection can varies by crop and study design due to several environmental factors such as occluding leaves and wind movement which can generate SfM artifacts, as well as changes based on growth stage, planting density, and image resolution [47, 49, 51]. The PlantHeightR app can easily handle these parameter optimizations, by providing common SfM metrics including the extraction methods mean, median and percentiles, as well as test different thresholds to vegetation and soil classification (Additional file 1: Table S1).

CONCLUSION

This study demonstrates the potential of using low-cost aerial images and advanced machine learning techniques, alongside well-established traditional methods, for HTP applications in dry bean trials. The developed pipelines provide plant breeders and researchers with a cost-effective and efficient solution by incorporating RM prediction, SC assessment, and PH estimation. Additionally, open-source software available in this study can support breeding decision-making and streamline HTP pipelines through user-friendly parameter analysis. CNN-LSTM and the Faster R-CNN models for predicting RM and SC, respectively, showed promising results in various flight conditions using DL applications, making them valuable tools for decision-making in breeding programs. The choice of image size, flight frequency, and the inclusion of GDD can have varying effects on the CNN-LSTM model performance depending on the environment. Future research should focus on refining these parameters to optimize RM model performance and further explore the potential of incorporating additional environmental factors, such as soil conditions and weather, to improve the accuracy and robustness of the model. Traditional methods for RM and SC analysis offer reliable alternatives to more complex deep learning techniques, although their success depends on optimal threshold selection and effective soil segmentation. Deep learning models demonstrated accurate performance in predicting RM with fewer flights and lower image resolution, as well as effective SC prediction in high-density crop populations. The assessment of plant distribution in the field is of great importance for optimizing crop management, improving yield predictions, and ultimately increasing grain yield. These advances in deep learning technology help reduce phenotypic errors caused by individual subjectivity differences, leading to more stable and reliable measurements with significant implications in plant breeding. Future research should focus on refining the DL models and exploring additional factors, such as plant health, weather conditions, and management practices, to improve

the accuracy and robustness of the predictions. Our findings also suggest that the optimum soil elevation data into PH estimation can lead to improved correlation coefficients, as well as highlight the importance of selecting the appropriate data source for PH estimation at dry beans across different environments. Further research is needed to investigate the potential of integrating multiple data sources, such as CSM/DSM and PC, and leveraging advanced machine learning algorithms to improve the accuracy and robustness of PH estimation across diverse crop species and growth stages.

DECLARATIONS

ACKNOWLEDGEMENTS

The authors wish to acknowledge the grower-cooperators who hosted these trials on their farms. We also thank Scott Bales for managing the Huron field site.

AUTHORS' CONTRIBUTIONS

LV, EW and FG devised and developed the methodology of experiments, and collected the ground notes, LV and EW collected the drone images, LV developed the image analysis algorithm, performed the statistical analysis, developed the software, and prepared the original manuscript, EW and FG critically reviewed and edited the draft, FG and EW administered the project, FG obtained the funding to support this study. All authors read and approved the final manuscript.

COMPETING INTERESTS

The authors declare that they have no competing interests.

AVAILABILITY OF DATA AND MATERIALS

The software developed are available in the GitHub repositories at https://github.com/msudrybeanbreeding and datasets generated and analyzed to perform RM, SC and PH pipelines during the current study are available online to download [138 – 140] at Zenodo deposit page (https://zenodo.org/).

CONSENT FOR PUBLICATION

All authors of the manuscript have read and agreed to its content and are accountable for all aspects of the accuracy and integrity of the manuscript.

REFERENCES

1. Uebersax MA, Cichy KA, Gomez FE, Porch TG, Heitholt J, Osorno JM, et al. Dry beans (Phaseolus vulgaris L.) as a vital component of sustainable agriculture and food security—A review. Legume Science [Internet]. 2022 [cited 2022 Jun 28]; Available from: https://onlinelibrary.wiley.com/doi/10.1002/leg3.155
2. USDA-NASS. Crop Values Summary 2021. 2022;48.
3. Heilig JA, Wright EM, Kelly JD. Symbiotic Nitrogen Fixation of Black and Navy Bean under Organic Production Systems. Agronomy Journal. 2017;109:2223–30.
4. Kelly JD, Kolkman JM, Schneider K. Breeding for yield in dry bean (Phaseolus vulgaris L.). Euphytica. 1998;102:14.
5. 2020 Michigan dry bean production survey [Internet]. Michigan Bean Commission. 2020 [cited 2023 May 1]. Available from: https://michiganbean.com/https-michiganbean-com-wp-content-uploads-2020-01-2019-michigan-dry-bean-survey-results-pdf-copy/
6. Schneider KA, Grafton KF, Kelly JD. QTL Analysis of Resistance to Fusarium Root Rot in Bean. Crop Sci. 2001;41:535–42.
7. White mold management in Michigan dry beans [Internet]. Dry Beans. 2020 [cited 2023 May 1]. Available from: https://www.canr.msu.edu/news/white-mold-management-in-michigan-dry-beans
8. Schwartz HF, Casciano DH, Asenga JA, Wood DR. Field Measurement of White Mold Effects upon Dry Beans with Genetic Resistance or Upright Plant Architecture [1]. Crop Sci. 1987;27:699–702.

9. Kelly JD, Adams MW. Phenotypic recurrent selection in ideotype breeding of pinto beans. Euphytica. 1987;36:69–80.
10. Zang J, Jin S, Zhang S, Li Q, Mu Y, Li Z, et al. Field-measured canopy height may not be as accurate and heritable as believed: evidence from advanced 3D sensing. Plant Methods. 2023;19:39.
11. Endres G, Crawford E, Franzen D, Hellevang K, Ikley J, Kandel H, et al. DRY BEAN Production Guide. Published in cooperation with and support from the Northarvest Bean Growers Association. 2019;A1133-20:128.
12. Singh D, Wang X, Kumar U, Gao L, Noor M, Imtiaz M, et al. High-Throughput Phenotyping Enabled Genetic Dissection of Crop Lodging in Wheat. Front Plant Sci. 2019;10:394.
13. Ninomiya S. High-throughput field crop phenotyping: current status and challenges. Breed Sci. 2022;72:3–18.
14. Araus JL, Kefauver SC, Zaman-Allah M, Olsen MS, Cairns JE. Translating High-Throughput Phenotyping into Genetic Gain. Trends in Plant Science. 2018;23:451–66.
15. Araus JL, Cairns JE. Field high-throughput phenotyping: the new crop breeding frontier. Trends in Plant Science. 2014;19:52–61.
16. Yang G, Liu J, Zhao C, Li Z, Huang Y, Yu H, et al. Unmanned Aerial Vehicle Remote Sensing for Field-Based Crop Phenotyping: Current Status and Perspectives. Front Plant Sci. 2017;8:1111.
17. Christenson BS, Schapaugh WT, An N, Price KP, Prasad V, Fritz AK. Predicting Soybean Relative Maturity and Seed Yield Using Canopy Reflectance. Crop Science. 2016;56:625–43.
18. Narayanan B, Floyd B, Tu K, Ries L, Hausmann N. Improving soybean breeding using UAS measurements of physiological maturity. In: Thomasson JA, McKee M, Moorhead RJ, editors. Autonomous Air and Ground Sensing Systems for Agricultural Optimization and Phenotyping IV [Internet]. Baltimore, United States: SPIE; 2019 [cited 2023 May 1]. p. 27. Available from: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/11008/2519072/Improving-soybean-breeding-using-UAS-measurements-of-physiological-maturity/10.1117/12.2519072.full
19. Volpato L, Dobbels A, Borem A, Lorenz AJ. Optimization of temporal UAS-based imagery analysis to estimate plant maturity date for soybean breeding. The Plant Phenome Journal [Internet]. 2021 [cited 2023 Jan 22];4. Available from: https://onlinelibrary.wiley.com/doi/10.1002/ppj2.20018
20. Litjens G, Kooi T, Bejnordi BE, Setio AAA, Ciompi F, Ghafoorian M, et al. A survey on deep learning in medical image analysis. Medical Image Analysis. 2017;42:60–88.
21. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Commun ACM. 2017;60:84–90.
22. Turaga SC, Murray JF, Jain V, Roth F, Helmstaedter M, Briggman K, et al. Convolutional Networks Can Learn to Generate Affinity Graphs for Image Segmentation. Neural Computation. 2010;22:511–38.
23. Hochreiter S, Schmidhuber J. Long Short-Term Memory. Neural Computation. 1997;9:1735–80.
24. Moeinizade S, Pham H, Han Y, Dobbels A, Hu G. An applied deep learning approach for estimating soybean relative maturity from UAV imagery to aid plant breeding decisions. Machine Learning with Applications. 2022;7:100233.
25. Trevisan R, Pérez O, Schmitz N, Diers B, Martin N. High-Throughput Phenotyping of Soybean Maturity Using Time Series UAV Imagery and Convolutional Neural Networks. Remote Sensing. 2020;12:3617.
26. Zhang S, Feng H, Han S, Shi Z, Xu H, Liu Y, et al. Monitoring of Soybean Maturity Using UAV Remote Sensing and Deep Learning. Agriculture. 2022;13:110.
27. Maimaitijiang M, Sagan V, Sidike P, Hartling S, Esposito F, Fritschi FB. Soybean yield prediction from UAV using multimodal data fusion and deep learning. Remote Sensing of Environment. 2020;237:111599.
28. Wang Y-H, Su W-H. Convolutional Neural Networks in Computer Vision for Grain Crop Phenotyping: A Review. Agronomy. 2022;12:2659.
29. Li L, Hassan MA, Yang S, Jing F, Yang M, Rasheed A, et al. Development of image-based wheat spike counter through a Faster R-CNN algorithm and application for genetic studies. The Crop Journal. 2022;10:1303–11.
30. Wang M, Fu B, Fan J, Wang Y, Zhang L, Xia C. Sweet potato leaf detection in a natural scene based on faster R-CNN with a visual attention mechanism and DIoU-NMS. Ecological Informatics. 2023;73:101931.

31. Zenkl R, Timofte R, Kirchgessner N, Roth L, Hund A, Van Gool L, et al. Outdoor Plant Segmentation With Deep Learning for High-Throughput Field Phenotyping on a Diverse Wheat Dataset. Front Plant Sci. 2022;12:774068.
32. David E, Daubige G, Joudelat F, Burger P, Comar A, De Solan B, et al. Plant detection and counting from high-resolution RGB images acquired from UAVs: comparison between deep-learning and handcrafted methods with application to maize, sugar beet, and sunflower [Internet]. Plant Biology; 2021 Apr. Available from: http://biorxiv.org/lookup/doi/10.1101/2021.04.27.441631
33. Velumani K, Lopez-Lozano R, Madec S, Guo W, Gillet J, Comar A, et al. Estimates of Maize Plant Density from UAV RGB Images Using Faster-RCNN Detection Model: Impact of the Spatial Resolution. Plant Phenomics. 2021;2021:2021/9824843.
34. Osco LP, Dos Santos De Arruda M, Gonçalves DN, Dias A, Batistoti J, De Souza M, et al. A CNN approach to simultaneously count plants and detect plantation-rows from UAV imagery. ISPRS Journal of Photogrammetry and Remote Sensing. 2021;174:1–17.
35. Wang P, Meng F, Donaldson P, Horan S, Panchy NL, Vischulis E, et al. High-throughput measurement of plant fitness traits with an object detection method using Faster R-CNN. New Phytologist. 2022;234:1521–33.
36. Beucher S, Meyer F. The Morphological Approach to Segmentation: The Watershed Transformation. In: Dougherty ER, editor. Mathematical Morphology in Image Processing [Internet]. 1st ed. CRC Press; 2018 [cited 2023 May 2]. p. 433–81. Available from: https://www.taylorfrancis.com/books/9781482277234/chapters/10.1201/9781482277234-12
37. Van Der Walt S, Schönberger JL, Nunez-Iglesias J, Boulogne F, Warner JD, Yager N, et al. scikit-image: image processing in Python. PeerJ. 2014;2:e453.
38. Bradski G. The OpenCV library. Dr Dobb's Journal of Software Tools. 2000;
39. Varela S, Dhodda P, Hsu W, Prasad PV, Assefa Y, Peralta N, et al. Early-Season Stand Count Determination in Corn via Integration of Imagery from Unmanned Aerial Systems (UAS) and Supervised Learning Techniques. Remote Sensing. 2018;10:343.
40. Kentsch S, Lopez Caceres ML, Serrano D, Roure F, Diez Y. Computer Vision and Deep Learning Techniques for the Analysis of Drone-Acquired Forest Images, a Transfer Learning Study. Remote Sensing. 2020;12:1287.
41. Sun S, Li C, Paterson A. In-Field High-Throughput Phenotyping of Cotton Plant Height Using LiDAR. Remote Sensing. 2017;9:377.
42. Han L, Yang G, Dai H, Yang H, Xu B, Feng H, et al. Fuzzy Clustering of Maize Plant-Height Patterns Using Time Series of UAV Remote-Sensing Images and Variety Traits. Front Plant Sci. 2019;10:926.
43. Castilla G, Filiatrault M, McDermid GJ, Gartrell M. Estimating Individual Conifer Seedling Height Using Drone-Based Image Point Clouds. Forests. 2020;11:924.
44. Wang Y, Chen Y. Non-Destructive Measurement of Three-Dimensional Plants Based on Point Cloud. Plants. 2020;9:571.
45. Tirado SB, Hirsch CN, Springer NM. UAV-based imaging platform for monitoring maize growth throughout development. Plant Direct [Internet]. 2020 [cited 2023 May 8];4. Available from: https://onlinelibrary.wiley.com/doi/10.1002/pld3.230
46. Hassan MA, Yang M, Fu L, Rasheed A, Zheng B, Xia X, et al. Accuracy assessment of plant height using an unmanned aerial vehicle for quantitative genomic analysis in bread wheat. Plant Methods. 2019;15:37.
47. Madec S, Baret F, De Solan B, Thomas S, Dutartre D, Jezequel S, et al. High-Throughput Phenotyping of Plant Height: Comparing Unmanned Aerial Vehicles and Ground LiDAR Estimates. Front Plant Sci. 2017;8:2002.
48. Malambo L, Popescu SC, Murray SC, Putman E, Pugh NA, Horne DW, et al. Multitemporal field-based plant height estimation using 3D point clouds generated from small unmanned aerial systems high-resolution imagery. International Journal of Applied Earth Observation and Geoinformation. 2018;64:31–42.
49. Watanabe K, Guo W, Arai K, Takanashi H, Kajiya-Kanegae H, Kobayashi M, et al. High-Throughput Phenotyping of Sorghum Plant Height Using an Unmanned Aerial Vehicle and Its Application to Genomic Prediction Modeling. Front Plant Sci [Internet]. 2017 [cited 2023 May 8];8. Available from: http://journal.frontiersin.org/article/10.3389/fpls.2017.00421/full
50. Thompson A, Thorp K, Conley M, Elshikha D, French A, Andrade-Sanchez P, et al. Comparing Nadir and Multi-Angle View Sensor Technologies for Measuring in-Field Plant Height of Upland Cotton. Remote Sensing. 2019;11:700.

51. Bendig J, Bolten A, Bennertz S, Broscheit J, Eichfuss S, Bareth G. Estimating Biomass of Barley Using Crop Surface Models (CSMs) Derived from UAV-Based RGB Imaging. Remote Sensing. 2014;6:10395–412.
52. Kawamura K, Asai H, Yasuda T, Khanthavong P, Soisouvanh P, Phongchanmixay S. Field phenotyping of plant height in an upland rice field in Laos using low-cost small unmanned aerial vehicles (UAVs). Plant Production Science. 2020;23:452–65.
53. Sweet DD, Tirado SB, Springer NM, Hirsch CN, Hirsch CD. Opportunities and challenges in phenotyping row crops using drone-based RGB imaging. The Plant Phenome Journal [Internet]. 2022 [cited 2023 May 2];5. Available from: https://onlinelibrary.wiley.com/doi/10.1002/ppj2.20044
54. Volpato L, Pinto F, González-Pérez L, Thompson IG, Borém A, Reynolds M, et al. High Throughput Field Phenotyping for Plant Height Using UAV-Based RGB Imagery in Wheat Breeding Lines: Feasibility and Validation. Front Plant Sci. 2021;12:591587.
55. Muasya RM, Lommen WJM, Muui CW, Struik PC. How weather during development of common bean (*Phaseolus vulgaris* L.) affects the crop's maximum attainable seed quality. NJAS: Wageningen Journal of Life Sciences. 2008;56:85–100.
56. Rife TW, Poland JA. Field Book: An Open-Source Application for Field Data Collection on Android. Crop Sci. 2014;54:1624–7.
57. R Core Team. R: A language and environment for statistical computing [Internet]. Vienna, Austria; 2023. Available from: https://www.R-project.org/
58. QGIS Development Team. QGIS geographic information system [Internet]. QGIS Association; 2023. Available from: https://www.qgis.org
59. Dutta A, Zisserman A. The VIA Annotation Software for Images, Audio and Video. Proceedings of the 27th ACM International Conference on Multimedia [Internet]. Nice France: ACM; 2019 [cited 2023 May 2]. p. 2276–9. Available from: https://dl.acm.org/doi/10.1145/3343031.3350535
60. Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans Pattern Anal Mach Intell. 2017;39:1137–49.
61. Pugh NA, Thorp KR, Gonzalez EM, Elshikha DEM, Pauli D. Comparison of image georeferencing strategies for agricultural applications of small unoccupied aircraft systems. The Plant Phenome Journal [Internet]. 2021 [cited 2023 May 2];4. Available from: https://onlinelibrary.wiley.com/doi/10.1002/ppj2.20026
62. Ivelja T, Bechor B, Hasan O, Miko S, Sivan D, Brook A. IMPROVING VERTICAL ACCURACY OF UAV DIGITAL SURFACE MODELS BY INTRODUCING TERRESTRIAL LASER SCANS ON A POINT-CLOUD LEVEL. Int Arch Photogramm Remote Sens Spatial Inf Sci. 2020;XLIII-B1-2020:457–63.
63. Huber PJ. Robust Estimation of a Location Parameter. In: Kotz S, Johnson NL, editors. Breakthroughs in Statistics [Internet]. New York, NY: Springer New York; 1992 [cited 2023 May 2]. p. 492–518. Available from: http://link.springer.com/10.1007/978-1-4612-4380-9_35
64. Biewald L. Experiment tracking with weights and biases [Internet]. 2020. Available from: https://www.wandb.com/
65. Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks.
66. Kingma DP, Ba J. Adam: A Method for Stochastic Optimization. arXiv; 2014 [cited 2023 May 2]; Available from: https://arxiv.org/abs/1412.6980
67. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine Learning in Python. arXiv; 2012 [cited 2023 May 3]; Available from: https://arxiv.org/abs/1201.0490
68. Cleveland WS, Devlin SJ. Locally Weighted Regression: An Approach to Regression Analysis by Local Fitting. Journal of the American Statistical Association. 1988;83:596–610.
69. MSU Dry Bean Breeding Program. msudrybeanbreeding/matuRity: matuRity app [Internet]. Zenodo; 2023 [cited 2023 May 10]. Available from: https://zenodo.org/record/7921260
70. MSU Dry Bean Breeding Program. msudrybeanbreeding/Vegetation_index_calculator: Vegetation index calculator [Internet]. Zenodo; 2023 [cited 2023 May 10]. Available from: https://zenodo.org/record/7921154
71. Rajeshwari P, Abhishek P, Vinod PS| T, Anurag Group of Institutions, Telangana, India. Object Detection: An Overview. IJTSRD. 2019;Volume-3:1663–5.
72. Girshick R. Fast R-CNN. 2015 IEEE International Conference on Computer Vision (ICCV) [Internet]. Santiago, Chile: IEEE; 2015 [cited 2023 May 3]. p. 1440–8. Available from: http://ieeexplore.ieee.org/document/7410526/

73. Cai Z, Fan Q, Feris RS, Vasconcelos N. A Unified Multi-scale Deep Convolutional Neural Network for Fast Object Detection. In: Leibe B, Matas J, Sebe N, Welling M, editors. Computer Vision – ECCV 2016 [Internet]. Cham: Springer International Publishing; 2016 [cited 2023 May 3]. p. 354–70. Available from: http://link.springer.com/10.1007/978-3-319-46493-0_22
74. He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) [Internet]. Las Vegas, NV, USA: IEEE; 2016 [cited 2023 May 3]. p. 770–8. Available from: http://ieeexplore.ieee.org/document/7780459/
75. Wang T, Zhao L, Li B, Liu X, Xu W, Li J. Recognition and counting of typical apple pests based on deep learning. Ecological Informatics. 2022;68:101556.
76. Kim K, Lee HS. Probabilistic Anchor Assignment with IoU Prediction for Object Detection [Internet]. arXiv; 2020 [cited 2023 May 3]. Available from: http://arxiv.org/abs/2007.08103
77. Buslaev A, Iglovikov VI, Khvedchenya E, Parinov A, Druzhinin M, Kalinin AA. Albumentations: Fast and Flexible Image Augmentations. Information. 2020;11:125.
78. Sharma DK. Information Measure Computation and its Impact in MI COCO Dataset. 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS) [Internet]. Coimbatore, India: IEEE; 2021 [cited 2023 May 3]. p. 1964–9. Available from: https://ieeexplore.ieee.org/document/9441788/
79. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521:436–44.
80. David E. Plant detection and counting from high-resolution RGB images acquired from UAVs: comparison between deep-learning and handcrafted methods with application to maize, sugar beet, and sunflower crops.
81. Matias FI, Caraza-Harter MV, Endelman JB. FIELDimageR: An R package to analyze orthomosaic images from agricultural field trials. Plant phenome j [Internet]. 2020 [cited 2022 Nov 27];3. Available from: https://onlinelibrary.wiley.com/doi/10.1002/ppj2.20005
82. Ronneberger O, Fischer P, Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation [Internet]. arXiv; 2015 [cited 2023 May 3]. Available from: http://arxiv.org/abs/1505.04597
83. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks [Internet]. arXiv; 2019 [cited 2023 May 1]. Available from: http://arxiv.org/abs/1801.04381
84. Adams J, Qiu Y, Xu Y, Schnable JC. Plant segmentation by supervised machine learning methods. Plant phenome j [Internet]. 2020 [cited 2023 May 6];3. Available from: https://onlinelibrary.wiley.com/doi/10.1002/ppj2.20001
85. Ulku I, Akagunduz E, Ghamisi P. Deep Semantic Segmentation of Trees Using Multispectral Images. IEEE J Sel Top Appl Earth Observations Remote Sensing. 2022;15:7589–604.
86. Smith AG, Petersen J, Selvan R, Rasmussen CR. Segmentation of roots in soil with U-Net. Plant Methods. 2020;16:13.
87. Snavely N, Seitz SM, Szeliski R. Modeling the World from Internet Photo Collections. Int J Comput Vis. 2008;80:189–210.
88. Ullman, S. The interpretation of structure from motion. Proc R Soc Lond B. 1979;203:405–26.
89. Schönberger JL, Zheng E, Frahm J-M, Pollefeys M. Pixelwise View Selection for Unstructured Multi-View Stereo. In: Leibe B, Matas J, Sebe N, Welling M, editors. Computer Vision – ECCV 2016 [Internet]. Cham: Springer International Publishing; 2016 [cited 2023 May 3]. p. 501–18. Available from: http://link.springer.com/10.1007/978-3-319-46487-9_31
90. Bolitho M, Kazhdan M, Burns R, Hoppe H. Parallel Poisson Surface Reconstruction. In: Bebis G, Boyle R, Parvin B, Koracin D, Kuno Y, Wang J, et al., editors. Advances in Visual Computing [Internet]. Berlin, Heidelberg: Springer Berlin Heidelberg; 2009 [cited 2023 May 3]. p. 678–89. Available from: http://link.springer.com/10.1007/978-3-642-10331-5_63
91. Susanto F, De Souza P, He J. Spatiotemporal Interpolation for Environmental Modelling. Sensors. 2016;16:1245.
92. Pix4D. How to improve the outputs of dense vegetation areas using PIX4Dmapper? [Internet]. Support. 2020 [cited 2023 May 3]. Available from: https://support.pix4d.com/hc/en-us/articles/202560159-How-to-improve-the-outputs-of-dense-vegetation-areas-using-PIX4Dmapper-
93. MSU Dry Bean Breeding Program. msudrybeanbreeding/PlantHeightR: PlantHeightR app [Internet]. Zenodo; 2023 [cited 2023 May 10]. Available from: https://zenodo.org/record/7921266
94. Ahmad S. On Efficient Monitoring of Process Dispersion using Interquartile Range. OJAppS. 2012;02:39–43.

95. Su J, Zhu X, Li S, Chen W-H. AI meets UAVs: A survey on AI empowered UAV perception systems for precision agriculture. Neurocomputing. 2023;518:242–70.
96. Zhang Z, Pope M, Shakoor N, Pless R, Mockler TC, Stylianou A. Comparing Deep Learning Approaches for Understanding Genotype × Phenotype Interactions in Biomass Sorghum. Front Artif Intell. 2022;5:872858.
97. Yasrab R, Pound MP, French AP, Pridmore TP. PhenomNet: Bridging Phenotype-Genotype Gap: A CNN-LSTM Based Automatic Plant Root Anatomization System [Internet]. Plant Biology; 2020 May. Available from: http://biorxiv.org/lookup/doi/10.1101/2020.05.03.075184
98. Taghavi Namin S, Esmaeilzadeh M, Najafi M, Brown TB, Borevitz JO. Deep phenotyping: deep learning for temporal phenotype/genotype classification. Plant Methods. 2018;14:66.
99. Luke JJ, Joseph R, Balaji M. IMPACT OF IMAGE SIZE ON ACCURACY AND GENERALIZATION OF CONVOLUTIONAL NEURAL NETWORKS. 2019;6.
100. Thambawita V, Strümke I, Hicks SA, Halvorsen P, Parasa S, Riegler MA. Impact of Image Resolution on Deep Learning Performance in Endoscopy Image Classification: An Experimental Study Using a Large Dataset of Endoscopic Images. Diagnostics. 2021;11:2183.
101. Shorten C, Khoshgoftaar TM. A survey on Image Data Augmentation for Deep Learning. J Big Data. 2019;6:60.
102. Blok PM, Kootstra G, Elghor HE, Diallo B, Van Evert FK, Van Henten EJ. Active learning with MaskAL reduces annotation effort for training Mask R-CNN on a broccoli dataset with visually similar classes. Computers and Electronics in Agriculture. 2022;197:106917.
103. Picek L, Šulc M, Patel Y, Matas J. Plant recognition by AI: Deep neural nets, transformers, and kNN in deep embeddings. Front Plant Sci. 2022;13:787527.
104. Sun C, Shrivastava A, Singh S, Gupta A. Revisiting Unreasonable Effectiveness of Data in Deep Learning Era. arXiv; 2017 [cited 2023 May 6]; Available from: https://arxiv.org/abs/1707.02968
105. Rasmussen CB, Kirk K, Moeslund TB. The Challenge of Data Annotation in Deep Learning—A Case Study on Whole Plant Corn Silage. Sensors. 2022;22:1596.
106. Zhang D, Han J, Cheng G, Yang M-H. Weakly Supervised Object Localization and Detection: A Survey. IEEE Trans Pattern Anal Mach Intell. 2021;1–1.
107. Li D, Zhu X, Song L. Mutual match for semi-supervised online evolutive learning. Appl Intell. 2023;53:3336–50.
108. Benato BC, Telea AC, Falcão AX. Iterative Pseudo-Labeling with Deep Feature Annotation and Confidence-Based Sampling. arXiv; 2021 [cited 2023 May 6]; Available from: https://arxiv.org/abs/2109.02717
109. Chebli A, Djebbar A, Marouani HF. Semi-Supervised Learning for Medical Application: A Survey. 2018 International Conference on Applied Smart Systems (ICASS) [Internet]. Medea, Algeria: IEEE; 2018 [cited 2023 May 6]. p. 1–9. Available from: https://ieeexplore.ieee.org/document/8651980/
110. Liao Y, Wu Y, Mo Y, Liu F, He Y, Zhao J. UPC-Faster-RCNN: A Dynamic Self-Labeling Algorithm for Open-Set Object Detection Based on Unknown Proposal Clustering. 2022 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI) [Internet]. Bedford, United Kingdom: IEEE; 2022 [cited 2023 May 6]. p. 1–6. Available from: https://ieeexplore.ieee.org/document/9913863/
111. Xiong L, Ye M, Zhang D, Gan Y, Liu Y. Source data-free domain adaptation for a faster R-CNN. Pattern Recognition. 2022;124:108436.
112. Ou J-R, Deng S-L, Yu J-G. WS-RCNN: Learning to Score Proposals for Weakly Supervised Instance Segmentation. Sensors. 2021;21:3475.
113. Liu J, Wang X. Plant diseases and pests detection based on deep learning: a review. Plant Methods. 2021;17:22.
114. Fuentes A, Yoon S, Lee MH, Park DS. Improving Accuracy of Tomato Plant Disease Diagnosis Based on Deep Learning With Explicit Control of Hidden Classes. Front Plant Sci. 2021;12:682230.
115. Karami A, Crawford M, Delp EJ. Automatic Plant Counting and Location Based on a Few-Shot Learning Technique. IEEE J Sel Top Appl Earth Observations Remote Sensing. 2020;13:5872–86.

116. Magoulianitis V, Ataloglou D, Dimou A, Zarpalas D, Daras P. Does Deep Super-Resolution Enhance UAV Detection? 2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS) [Internet]. Taipei, Taiwan: IEEE; 2019 [cited 2023 May 8]. p. 1–6. Available from: https://ieeexplore.ieee.org/document/8909865/
117. Haris M, Shakhnarovich G, Ukita N. Task-Driven Super Resolution: Object Detection in Low-Resolution Images. In: Mantoro T, Lee M, Ayu MA, Wong KW, Hidayanto AN, editors. Neural Information Processing [Internet]. Cham: Springer International Publishing; 2021 [cited 2023 May 8]. p. 387–95. Available from: https://link.springer.com/10.1007/978-3-030-92307-5_45
118. Ji H, Gao Z, Mei T, Li Y. Improved Faster R-CNN With Multiscale Feature Fusion and Homography Augmentation for Vehicle Detection in Remote Sensing Images. IEEE Geosci Remote Sensing Lett. 2019;16:1761–5.
119. Dong C, Loy CC, He K, Tang X. Image Super-Resolution Using Deep Convolutional Networks. IEEE Trans Pattern Anal Mach Intell. 2016;38:295–307.
120. Wang X, Yu K, Wu S, Gu J, Liu Y, Dong C, et al. ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks. In: Leal-Taixé L, Roth S, editors. Computer Vision – ECCV 2018 Workshops [Internet]. Cham: Springer International Publishing; 2019 [cited 2023 May 8]. p. 63–79. Available from: https://link.springer.com/10.1007/978-3-030-11021-5_5
121. Zhang K, Zuo W, Zhang L. Learning a Single Convolutional Super-Resolution Network for Multiple Degradations. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition [Internet]. Salt Lake City, UT: IEEE; 2018 [cited 2023 May 8]. p. 3262–71. Available from: https://ieeexplore.ieee.org/document/8578442/
122. Anwar S, Khan S, Barnes N. A Deep Journey into Super-resolution: A Survey. ACM Comput Surv. 2021;53:1–34.
123. He K, Gkioxari G, Dollar P, Girshick R. Mask R-CNN. IEEE Trans Pattern Anal Mach Intell. 2020;42:386–97.
124. Machefer M, Lemarchand F, Bonnefond V, Hitchins A, Sidiropoulos P. Mask R-CNN Refitting Strategy for Plant Counting and Sizing in UAV Imagery. Remote Sensing. 2020;12:3015.
125. PratimAcharjya P, Ghoshal D. Watershed Segmentation based on Distance Transform and Edge Detection Techniques. IJCA. 2012;52:6–10.
126. Xu C, Li R, Song W, Wu T, Sun S, Han T, et al. High Density and Uniform Plant Distribution Improve Soybean Yield by Regulating Population Uniformity and Canopy Light Interception. Agronomy. 2021;11:1880.
127. Benjamin LR. Growth Analysis, Crops. Encyclopedia of Applied Plant Sciences [Internet]. Elsevier; 2017 [cited 2023 May 8]. p. 23–8. Available from: https://linkinghub.elsevier.com/retrieve/pii/B9780123948076002252
128. Matsuo N, Yamada T, Takada Y, Fukami K, Hajika M. Effect of plant density on growth and yield of new soybean genotypes grown under early planting condition in southwestern Japan. Plant Production Science. 2018;21:16–25.
129. Wu L, Deng Z, Cao L, Meng L. Effect of plant density on yield and Quality of perilla sprouts. Sci Rep. 2020;10:9937.
130. Yang F, Huang S, Gao R, Liu W, Yong T, Wang X, et al. Growth of soybean seedlings in relay strip intercropping systems in relation to light quantity and red:far-red ratio. Field Crops Research. 2014;155:245–53.
131. Masino A, Rugeroni P, Borrás L, Rotundo JL. Spatial and temporal plant-to-plant variability effects on soybean yield. European Journal of Agronomy. 2018;98:14–24.
132. Liu X, Rahman T, Song C, Yang F, Su B, Cui L, et al. Relationships among light distribution, radiation use efficiency and land equivalent ratio in maize-soybean strip intercropping. Field Crops Research. 2018;224:91–101.
133. Zhang Y, Xu Z, Li J, Wang R. Optimum Planting Density Improves Resource Use Efficiency and Yield Stability of Rainfed Maize in Semiarid Climate. Front Plant Sci. 2021;12:752606.
134. Tokatlidis I. Crop resilience via inter-plant spacing brings to the fore the productive ideotype. Front Plant Sci. 2022;13:934359.
135. Li W, Niu Z, Chen H, Li D, Wu M, Zhao W. Remote estimation of canopy height and aboveground biomass of maize using high-resolution stereo images from a low-cost unmanned aerial vehicle system. Ecological Indicators. 2016;67:637–48.
136. James MR, Robson S. Straightforward reconstruction of 3D surfaces and topography with a camera: Accuracy and geoscience application: 3D SURFACES AND TOPOGRAPHY WITH A CAMERA. J Geophys Res. 2012;117:n/a-n/a.
137. Mao Z, Hu S, Wang N, Long Y. Precision Evaluation and Fusion of Topographic Data Based on UAVs and TLS Surveys of a Loess Landslide. Front Earth Sci. 2021;9:801293.
138. Volpato, Leonardo, Wright, Evan, Gomez, Francisco. Digital Phenotyping in Plant Breeding: Evaluating Relative Maturity, Stand Count, and Plant Height in Dry Beans via RGB Drone-Based Imagery and Deep Learning Approaches 2023.

https://doi.org/10.5281/zenodo.7922589.

139. Volpato, Leonardo, Wright, Evan, Gomez, Francisco. Digital Phenotyping in Plant Breeding: Evaluating Relative Maturity, Stand Count, and Plant Height in Dry Beans via RGB Drone-Based Imagery and Deep Learning Approaches 2023. https://doi.org/10.5281/zenodo.7922584.

140. Volpato, Leonardo, Wright, Evan, Gomez, Francisco. Digital Phenotyping in Plant Breeding: Evaluating Relative Maturity, Stand Count, and Plant Height in Dry Beans via RGB Drone-Based Imagery and Deep Learning Approaches 2023. https://doi.org/10.5281/zenodo.7922565

Figures

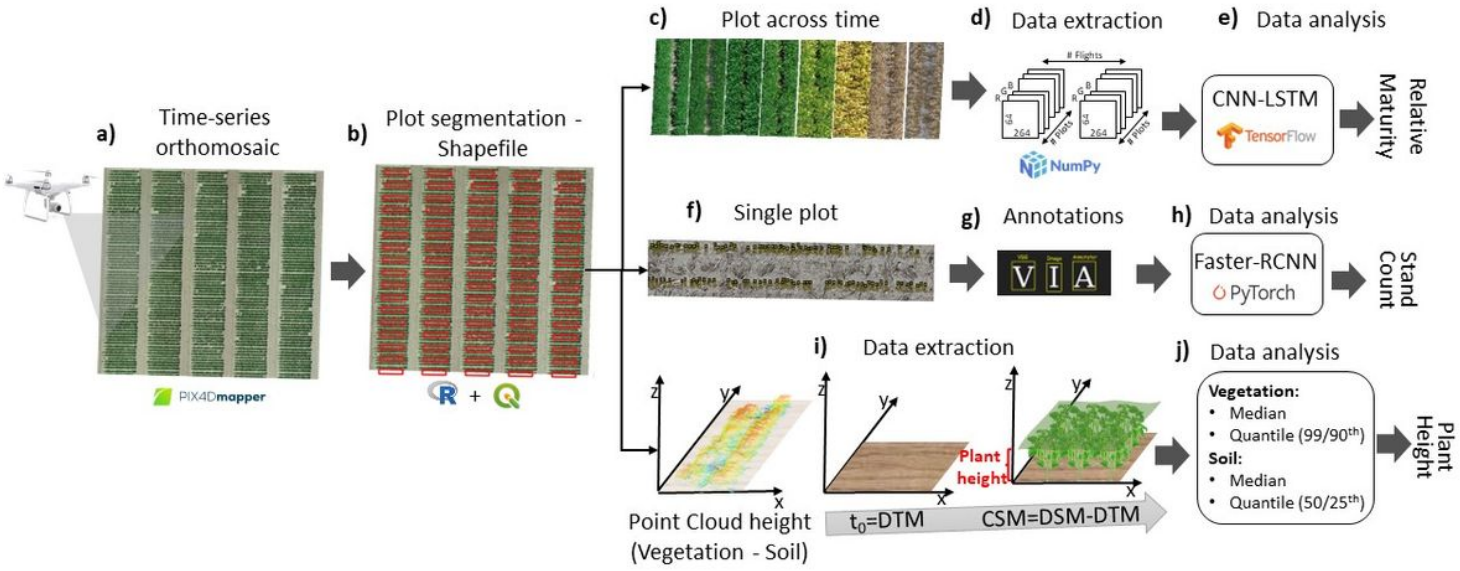

Figure 1

Illustration of the general HTP pipeline for relative maturity (RM), plant height (PH) and stand count (SC) estimations. a) Raw JPEG images were captured by an RGB high-resolution drone. Pix4Dmapper software was used to obtain orthomosaics, DSM and Point Cloud data. b) Shapefile was constructed using the QGIS software integrated with a plugin using R software. Maturity was predicted using a temporal image data collected across the season, and the images were transformed into a high dimension array using the Python library NumPy to finally use as input in the CNN-LSTM model using the TensorFlow library in python (*c* to *e* steps). Stand count was done using single plant annotations and trained using Faster-RCNN model in PyTorch (*f* to *h* steps). Plant height was obtained through DSM (or DTM for bare soil flight) and Point Cloud (vegetation and soil flight) performing the subtraction of the elevation from vegetation minus soil using median and quantile to perform pixels values extractions (*i* and *j*).

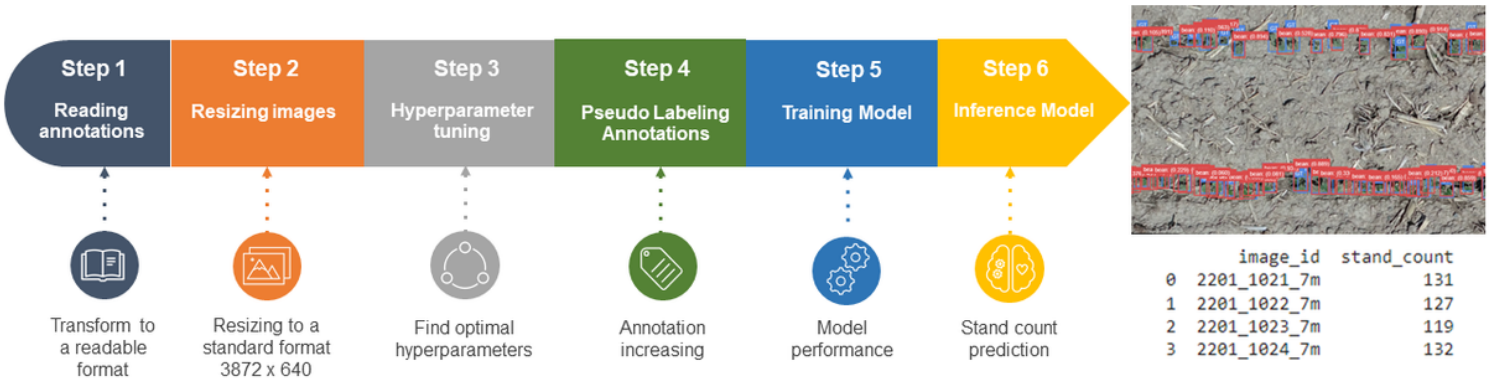

Figure 2

Stand count pipeline deployed to perform the object detection analysis using Faster-RCNN model. First, the annotations were converted to a readable format using the boxes edges (Step 1), and in the second step the images were resized to be ready to use in the training model. Step 3 aimed to find the optimized hyperparameters. Then, to increase the train data set, pseudo labeling annotations were performed using the non-annotated experimental plots repetitions 3 and 4 (Step 4). Finally, in Step 5 the object detection model was deployed using the optimized parameters and annotations (image annotations and pseudo annotations) to train the model that will be used to perform plant identification and counting in Step 6.

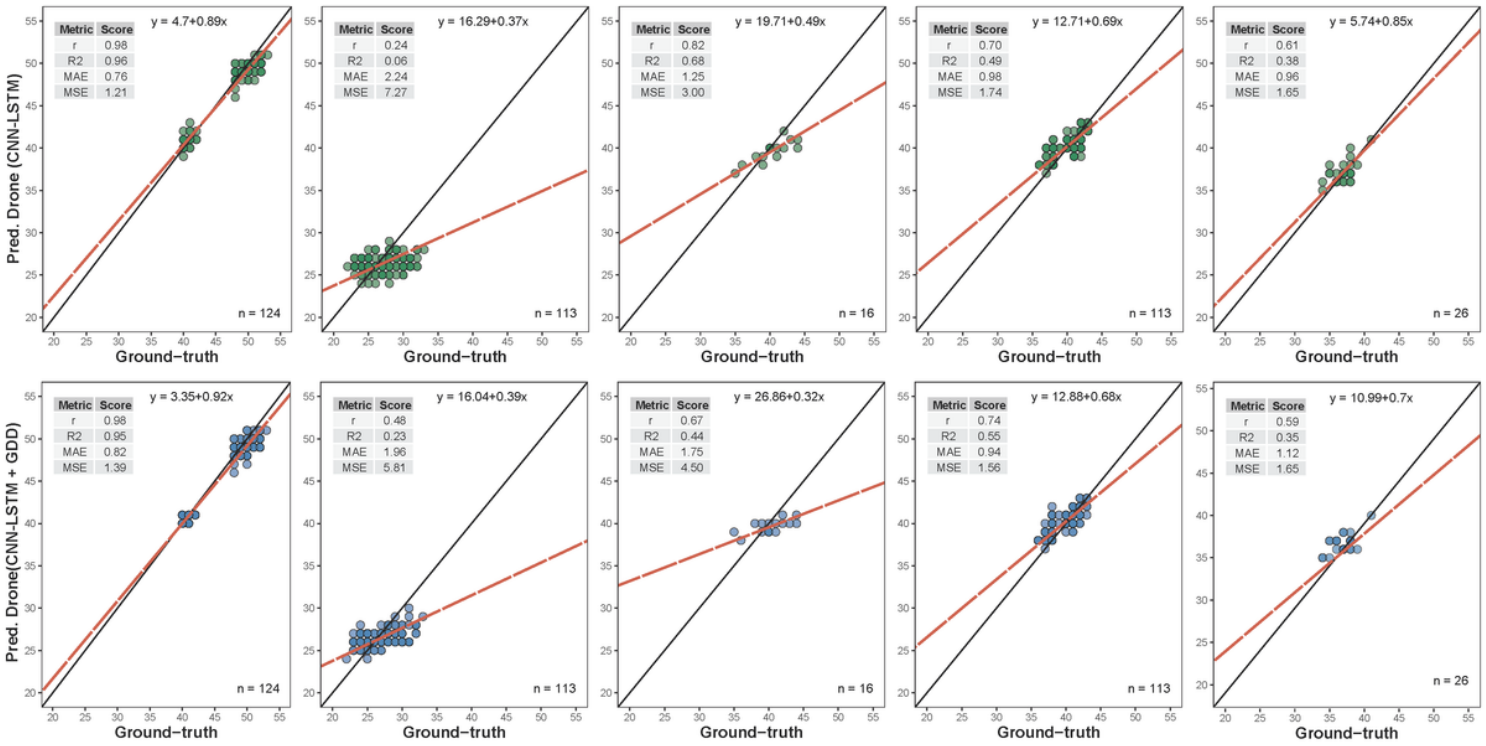

Figure 3

Performance comparison between the CNN-LSTM model and the CNN-LSTM + GDD model across five different environments using 6 flights and 256x64 image size dataset under four evaluation metrics: Pearson's correlation coefficient (*r*), coefficient of determination (r^2), Mean Absolute Error (*MAE*), and Mean Squared Error (*MSE*). *n* is the total number of observations in each environment.

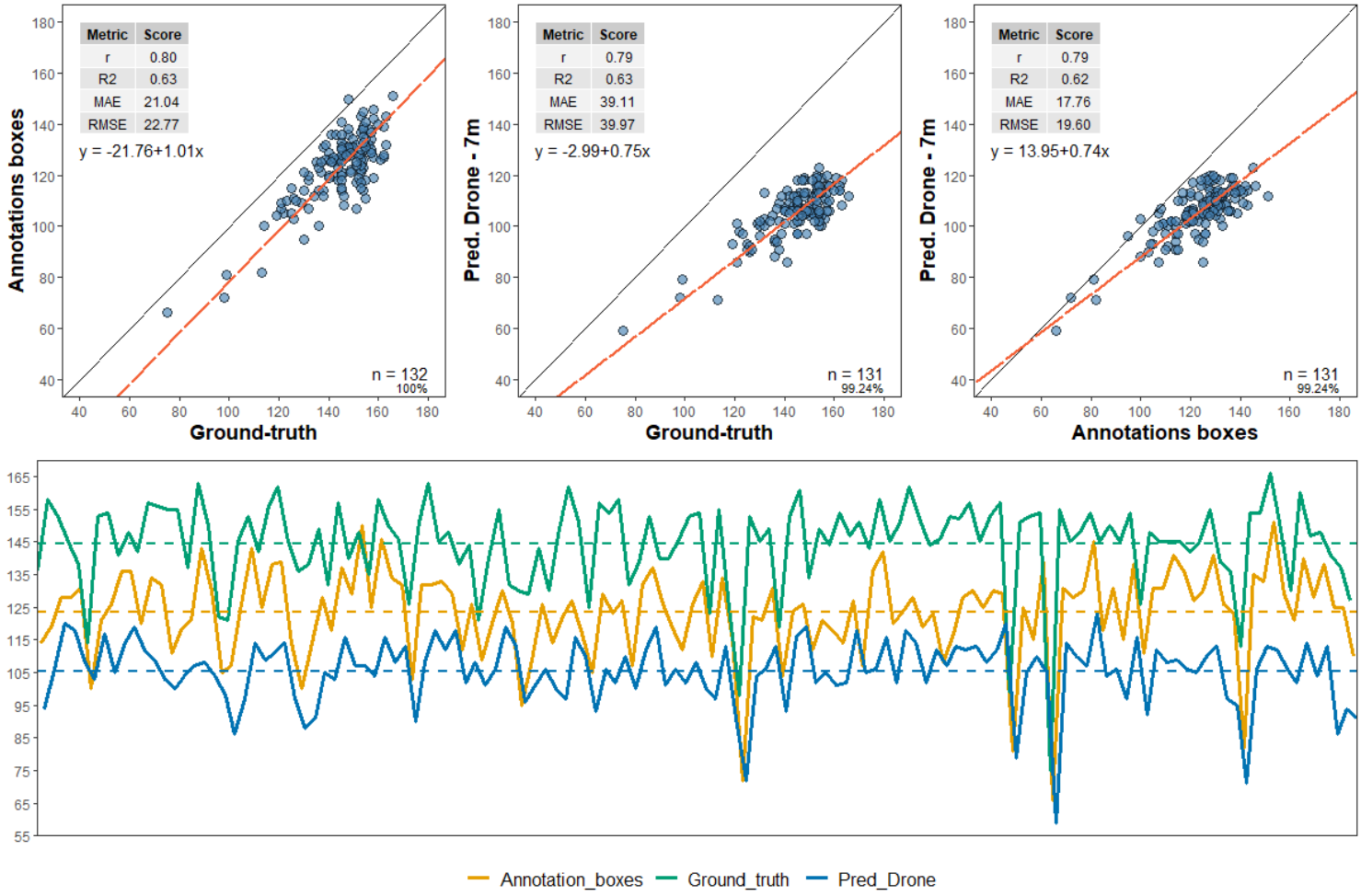

Figure 4

The upper figure presents the performance metrics for the annotation boxes compared to the ground-truth measurements (GT), as well as the predicted drone measurements at 7 meters, and the bottom shows the trend lines across measured plots to annotation boxes (orange), GT (green) and predicted by drone (blue) to stand count (SC).

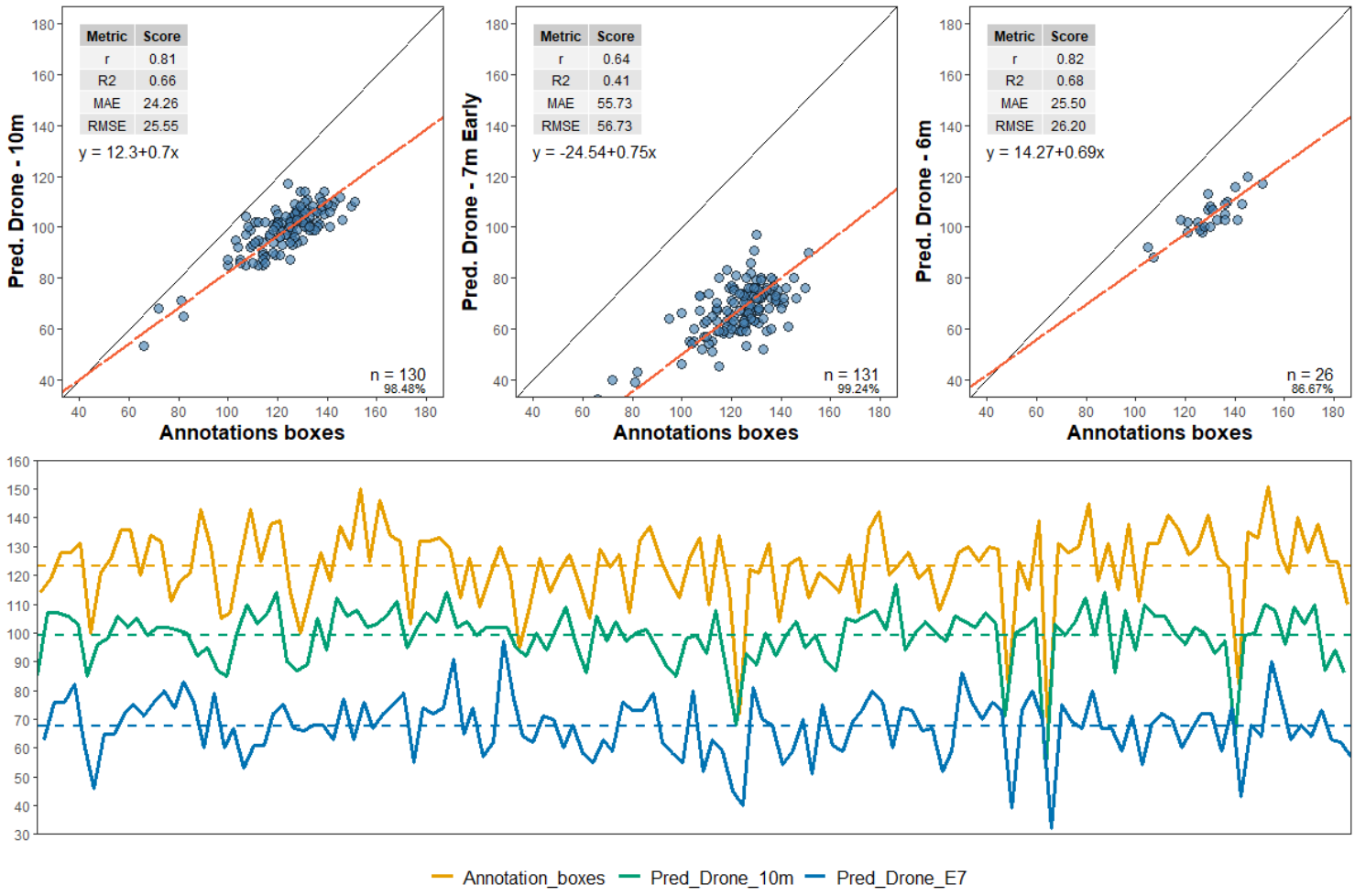

Figure 5

The upper figure shows the performance metrics for the predicted drone measurements at 10 meters, 7 meters early flight, as well as 6 meters compared to the annotation, and the bottom figure shows the trend lines across measured plots to annotation boxes (orange), predicted by drone at 10 meters flight (green), and predicted by drone at 7 meters early flight (blue) to stand count (SC).

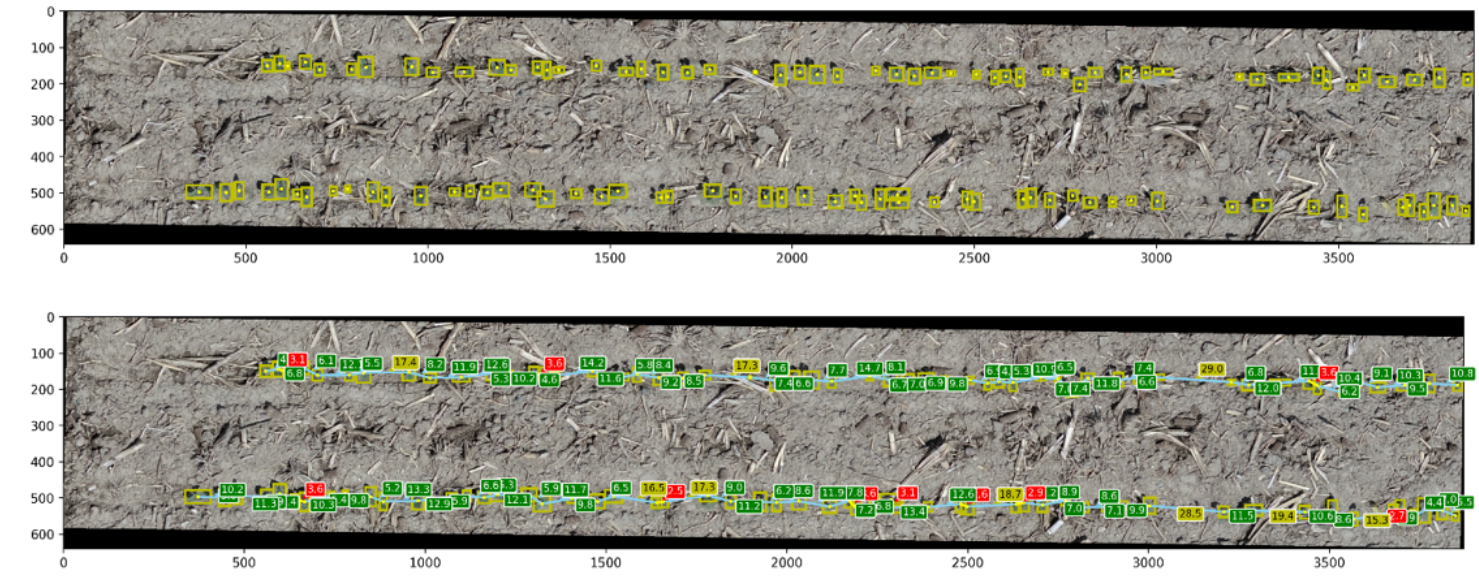

Figure 6

Spatial plant distribution (PD) within a dry bean field plot. The upper image presents the bean plant annotated with a centroid point and the lower image represents the distance among the annotated plants in centimeters (cm). In the lower image, red and yellow boxes depict a poor PD to nearby and distant plants, respectively, while green boxes show the optimal distribution of plants. The distance between plants is represented in cm calculated based on the pixel image size (GSD).

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- Suppl.zip