High throughput measurement of plant fitness traits with an object detection method using Faster R-CNN

Peipei Wang (orcid.org/0000-0002-7580-9627)[1,2‡], Fanrui Meng (orcid.org/0000-0002-7911-6991)[1,2‡], Paityn Donaldson[1], Sarah Horan[1], Nicholas L. Panchy (orcid.org/0000-0002-1551-3517)[3], Elyse Vischulis[4], Eamon Winship (orcid.org/0000-0002-4200-3694)[5], Jeffrey K. Conner (orcid.org/0000-0003-1613-5826)[1,6,7], Patrick J. Krysan (orcid.org/0000-0003-4916-915X)[8], Shin-Han Shiu (orcid.org/0000-0001-6470-235X)[1,2,4,7,9], Melissa D. Lehti-Shiu (orcid.org/0000-0003-1985-2687)[1]

[1]Department of Plant Biology, Michigan State University, East Lansing, MI, 48824, USA
[2]DOE Great Lake Bioenergy Research Center, Michigan State University, East Lansing, MI, 48824, USA,
[3]National Institute for Mathematical and Biological Synthesis, University of Tennessee, 1122 Volunteer Blvd., Suite 106, Knoxville, TN, 37996-3410, USA
[4]Genetics and Genome Sciences Graduate Program, Michigan State University, East Lansing, MI, 48824, USA
[5]Department Of Biochemistry and Molecular Biology, Michigan State University, East Lansing, MI, 48824, USA
[6]W.K. Kellogg Biological Station, Michigan State University, 3700 E. Gull Lake Drive, Hickory Corners, MI, 49060, USA.
[7]Ecology, Evolution, and Behavior Graduate Program, Michigan State University, East Lansing, MI, 48824, USA
[8]Department of Horticulture, University of Wisconsin-Madison, Madison, WI, 53705, USA
[9]Department of Computational Mathematics, Science, and Engineering, Michigan State University, East Lansing, MI 48824, USA

‡: Joint first authors

Brief heading: High throughput phenotyping of Arabidopsis fitness traits.

Authors for correspondence:

- *Shin-Han Shiu- Tel: +1-517-353-7196; Email: shius@msu.edu, website: https://shiulab.github.io/*
- *Melissa D. Lehti-Shiu: Tel: +1-517-884-4066: Email: lehtishi@msu.edu, website: https://shiulab.github.io/*

Total word count for the main body: 6108 words

Introduction: 1141 words
Materials and Methods: 980 words
Results: 2546 words
Discussion: 1441 words

There are:

7 figures, all of them should be published in color
9 supplementary figures

12 supplementary tables

Summary

- Revealing the contributions of genes to plant phenotype is frequently challenging because loss-of-function effects may be subtle or masked by varying degrees of genetic redundancy. Such effects can potentially be detected by measuring plant fitness, which reflects the cumulative effects of genetic changes over the lifetime of a plant. However, fitness is challenging to measure accurately, particularly in species with high fecundity and relatively small propagule sizes such as *Arabidopsis thaliana*.
- An image segmentation-based method using the software ImageJ and an object detection-based method using the Faster Region Based Convolutional Neural Network algorithm were used for measuring two Arabidopsis fitness traits: seed and fruit counts.
- The segmentation-based method was error-prone (correlation between true and predicted seed counts, r^2=0.849) because seeds touching each other were undercounted. In contrast, the object detection-based algorithm yielded near perfect seed counts (r^2=0.9996) and highly accurate fruit counts (r^2=0.980). Comparing seed counts for wild type and 12 mutant lines revealed fitness effects for three genes; fruit counts revealed the same effects for two genes.
- Our study provides analysis pipelines and models to facilitate the investigation of Arabidopsis fitness traits and demonstrates the importance of examining fitness traits when studying gene functions.

Keywords: fitness traits; deep learning; machine vision; segmentation; object detection; Arabidopsis

Introduction

A major goal of biology is to understand the molecular basis for the development of organisms and their adaptation to different environments (McDonald, 1983). One approach is to evaluate the effects of genetic variants on phenotypes. However, it is often challenging to investigate such effects because gene functions may be masked by genetic redundancy (Bouché & Bouchez, 2001; Sun *et al.*, 2012) and/or be condition specific (Hirsch *et al.*, 1998; Meissner *et al.*, 1999). Moreover, the physiological and/or developmental changes caused by loss of gene function may be too subtle to detect. This challenge can be alleviated by measuring the effects of genetic variations on fitness (i.e., the ability of an individual to survive and reproduce) because it reflects the cumulative effects of genetic changes over the lifetime of a plant. Accurate estimates of fitness are therefore valuable for several fields of study, including plant genetics, evolution, and plant breeding.

Among fitness measures, the most direct measure is the number of progenies produced (Thomson & Hadfield, 2017). In *Arabidopsis thaliana*, a predominantly selfing plant, the total number of seeds produced per plant is a particularly good estimate of fitness because it incorporates both male and female contributions. However, because Arabidopsis seeds are small (~0.1–0.2 mm^2; Jahnke *et al.*, 2016) and produced in large numbers (up to thousands per plant; Boyes *et al.*, 2001; Morales *et al.*, 2020), it is difficult to obtain accurate seed counts. As a consequence, fruit (silique) number (Busoms *et al.*, 2015) and total fruit length (Roux *et al.*, 2004; Kerwin *et al.*, 2015; Busoms *et al.*, 2015) are often used to measure fitness. Both measures are correlated with seed production, but fruit number is not perfectly correlated with seed number (e.g., r^2=0.960, Mauricio & Rausher, 1997) and correlations with fruit length are highly variable across studies, ranging from r^2=0.988 (Roux *et al.*, 2004) to r^2=0.256 (Gnan *et al.*, 2014). In addition, fruit numbers (up to 450 per plant; Hamidinekoo *et al.*, 2020) are typically counted manually, and these counts can be error prone. Thus, to better measure fitness, both fruit and seed numbers should be evaluated using methods that are not hindered by propagule size or number.

Several programs have been designed to increase the efficiency and accuracy of seed analyses. Some are aimed at measuring the properties of individual seeds (e.g., size and shape) and others at obtaining high throughput seed counts (Herridge *et al.*, 2011; Tanabata *et al.*, 2012; Moore *et al.*, 2013). These approaches typically require that seeds be separated before imaging, which increases the time needed for processing. Other systems have been designed to separate seeds mechanically such as the *pheno*Seeder device (Jahnke *et al.*, 2016), large-particle flow cytometer (Morales *et al.*, 2020), and the BELT imaging system combined with the phenoSEED algorithm (Halcro *et al.*, 2020). A drawback of these methods is that they require specialized equipment, hindering their widespread adoption. Another approach that has been increasingly used in plant biology for applications such as measurement of fitness traits is machine vision, the application of deep learning algorithms to image analysis (Mochida *et al.*, 2019).

Deep learning approaches, in particular Convolutional Neural Network (CNN)-based frameworks, have been developed to detect vastly different objects (from cars to plant seeds) in images. For example, aiming to train instance segmentation models where seed counting was not the primary task, Toda *et al.* (2020) were able to detect the seeds of rice, lettuce, oat, and wheat with 96% recall and 95% precision using Mask Region Based CNN (R-CNN). However, the detection of much smaller objects using CNN-based approaches remains challenging (Cao *et al.*, 2019), likely because CNNs create low-level abstractions of the images, and if the objects are too small, the resulting abstractions are too simple to be used to distinguish whether the object is present or not. Although the CNN-based models developed by Toda *et al.* (2020) detected seeds with high accuracy, the smallest seeds tested were lettuce seeds, which have areas ranging from 1.5–3.6 mm^2 (Penaloza *et al.*, 2005) and are ~10 times larger than Arabidopsis seeds. Another consideration is that the most convenient way to count all the seeds from an Arabidopsis plant, which can produce thousands of seeds (Boyes *et al.*, 2001; Morales *et al.*, 2020), would be to put all the seeds in a single image, thus resulting in a relatively small ratio of seed size to image size. However, because of the small images (1024 × 1024 px^2 or 2000 × 2000 px^2) used in Toda *et al.* (2020), the ratio of seed size to image size was relatively large (>5000 px^2 per barley seed), which limited the number of seeds that could be included in an image. Therefore, it is important to assess how well the CNN-based

approaches perform in detecting objects as small as Arabidopsis seeds in an image containing thousands of them.

CNN-based approaches have also been used in fruit counting. For example, wheat spikes can be detected, counted, and analyzed to estimate yield using R-CNN (correlation between true and predicted counts: r^2=0.93 with a slope of 1.01; Hasan *et al.*, 2018). Starting from two pre-trained models (ResNet and ResNext), Afonso *et al.*, (2020) applied the Mask R-CNN approach to detect and count tomato fruits from images, obtaining an F1 of 0.94 when fruits partially overlapped with each other. DeepPod effectively counts Arabidopsis fruits but results in a high number of false negatives when there are many fruits (r^2=0.90 with a slope of ~0.70; Hamidinekoo *et al.*, 2020). In addition, the inflorescences need to be harvested when the fruits are still green, preventing the harvesting of seeds for future propagation or analysis. Thus, it is important to develop tools or models to detect and count mature fruits when seeds need to be saved for future experiments. Because Arabidopsis fruits shatter easily when dry, such tools should ideally be able to count fruits at different stages, including intact fruits and those that have already dehisced and released seed.

In this study, we evaluated two approaches for counting seeds from an Arabidopsis plant in a single image: (1) a segmentation-based method using the software ImageJ (Schneider *et al.*, 2012) and (2) an object detection method using the Faster R-CNN algorithm (Ren *et al.*, 2017). We also applied Faster R-CNN to count fruits in whole plant images captured after seeds were mature. To facilitate seed and fruit counting in diverse images, we established models using input images with varying resolution, contrast, brightness, and blurriness. The final seed and fruit models are provided and can be readily used by the research community. Finally, we used our pipeline to count seeds for loss-of-function mutants of six pairs of duplicate genes. We showed that mutation of three genes affects fitness, illustrating the potential importance of measuring fitness traits and the utility of our pipeline in the investigation of gene functions.

Materials and Methods

Plant materials

T-DNA insertion mutants in the *Arabidopsis* Col-0 background and wild-type (WT) Col-0 controls were used for training seed and fruit counting models. Information about these lines is provided in **Tables S1**, **S2**, and **S3**. Fitness data are reported for T-DNA insertion mutants of *PURPLE ACID PHOSPHATASE 2* (*PAP2*), *PAP9*, *HIGH MOBILITY GROUP A4* (*HON4*), *HON5*, *EUKARYOTIC INITIATION FACTOR 4B1* (*EIF4B1*), *EIF4B2*, *ADENOSINE 5'-PHOSPHOSULFATE REDUCTASE-LIKE 5* (*APRL5*), *APRL7*, *PLANT AND FUNGI ATYPICAL DUAL-SPECIFICITY PHOSPHATASE 3* (*PFA-DSP3*), *PFA-DSP5*, *PFA-KINESIN 7.2* (*KIN7.2*), and *KIN7.4* (**Tables S4, S5**). These mutants were collected as part of a large-scale study to assess the degree of genetic redundancy between duplicate genes. Multiple homozygous mutant and WT sibling plants were identified by PCR with gene-specific primers (two to six plants per genotype, **Table S3**). Seeds harvested from these independent lines (referred to as sublines) were planted (n = 5–20 per subline, total $n \geq 40$ per genotype) for fitness comparison between mutants and WT, and each mutant was compared with its WT sibling. This was done to reduce the chance that observed fitness effects were due to other undetected T-DNA insertions.

For plants grown for fitness analysis (**Tables S3-S5**) and seed scan images (**Table S1**), seeds were grown as described in **Methods S1**. Plants were grown until they were mature (i.e., had undergone global arrest). When plants were completely dry, the number of intact and completely or partially shattered fruits from each plant were processed as detailed in **Methods S1**. The total seed number produced per plant was estimated in two steps. First, the seed number was divided by the number of intact fruits to obtain the average seed number per fruit. Second, the average seed number per fruit was multiplied by the total fruit number (both intact and shattered) to estimate the total seed number per plant. Plants used for fruit imaging (**Table S2**) were grown as described in **Methods S1**.

Seed image scanning, processing, and counting with the segmentation method

Prior to seed imaging, we separated the seeds from the chaff (see **Methods S1**). Seed images were obtained by placing petri plate lids containing seeds in a template made

from white acrylic (295 mm × 210 mm × 10 mm, **Fig. 1a**) and taking scans with a desktop scanner (see **Methods S1**). The ImageJ (version 1.52a, https://imagej.nih.gov, Schneider *et al.*, 2012) workflow is shown in **Fig. 1**. Details about seed counting using ImageJ are in **Methods S1**. The image conversion program and the ImageJ macro were combined into a Windows batch script (available in our Github repository, see **Data availability**), in which a for-loop was used to quickly count seeds for images in sequence. It took approximately 5 min to fully process 10 images.

Seed image processing and counting with an object detection method using Faster R-CNN

Before seed detection, each scanned image was split into 12 sub images; each sub image contains a single plate lid and is referred to as a "whole-plate image". After testing several algorithms, we chose to use Faster R-CNN for seed detection (for reasons, see **Methods S1**). Faster R-CNN combines the generation of region proposals (i.e., circumscribing the areas of interest, a regression problem) and their classification (i.e., in our case, the object is a seed or not) into a single pipeline (Ren *et al*., 2017). In Faster R-CNN, images were first processed by a feature extractor (Inception v.2; Szegedy *et al*., 2016), and the resulting feature maps were used to predict bounding boxes (referred to as proposals) containing images of individual seeds (left panel in **Fig. S1**); then these proposals were used to crop features from the feature maps (right panel in **Fig. S1**). These cropped features were subsequently used for classification and bounding box regression.

Faster R-CNN models were trained using Tensorflow object detection API (Huang *et al*., 2017) and implemented in Tensorflow v1.13.2 (Abadi *et al*., 2016) in python v3.6.4. In the initial Faster R-CNN modeling trial, each whole-plate image was split into four quarter-plate images. Images were pre-processed and seeds were annotated as detailed in **Methods S1**. To speed up the training process, a pre-trained model (faster_rcnn_inception_v2_coco) was used as a starting point. To optimize Arabidopsis seed detection, we conducted hyperparameter tuning (**Methods S1**, **Tables S6**, **S7**, and **Fig. S2**, **S3**) and evaluated tuned models using the measure IoU, which is defined as the

intersection (I) over (o) the union (U) of a ground truth area and a prediction area, as detailed in **Methods S1**.

Fruit image capturing and counting with an object detection method based on Faster R-CNN

Each dry Arabidopsis plant was placed on a pink paper background and photographed with an iPhone 8 smartphone. The images were saved in jpeg format with dimensions of 3024 × 4032 pixels. Fruits in the images were manually annotated, and the annotated coordinates were then converted to the csv and TFrecord formats, as conducted for the seed images (**Methods S1**). The same pre-trained Faster R-CNN model used for seed counting was used to build the fruit counting models, and the same three hyperparameters were tuned to optimize the model performance but with a different hyperparameter space (**Table S8**). For each hyperparameter combination, a model was saved after 6000 steps, when the performance had converged. A final model was established using hyperparameters selected based on performance on the validation set images.

Statistical analysis of fitness traits

Data from the border cells (see **Methods S1**) showed different distributions compared with data from inside cells; therefore, these data were excluded from further analysis. For each block (i.e., one including *pap*, *hon*, and *eif4b* and one including *aprl*, *pfa-dsp*, and *kin7*, see **Methods S1**), quantile normalization was performed across flats using R package "broman" (https://github.com/kbroman/broman) to account for variation between flats. Each mutant was compared with its WT control using the Wilcoxon rank-sum test. Each pair of duplicate genes had the same WT sibling control.

Results

Seed counting with the segmentation method using ImageJ

Because ImageJ is widely used for seed morphology analysis (Cervantes *et al.*, 2016), we first developed a pipeline for seed counting that incorporated ImageJ analysis based on segmentation of seed areas. When fewer than 200 seeds were placed on the plate lid and separated using forceps, seeds were detected and counted with high accuracy (correlation between true and predicted seed counts, r^2=0.996, slope=0.9998, 60 images, **Fig. 1b,c, Table S9**). Our segmentation-based pipeline allowed the detection of about 52 template images (total of 624 plate lids) per hour with a typical laptop (Intel(R) Core i7-7500U CPU, 16GB RAM).

However, when seeds were placed on plate lids without separation, big clumps of seeds were not counted by the segmentation method, and small clumps where a small number of seeds were touching each other were recognized as single seeds (**Fig. 2a**). The prediction accuracy drops off as the number of seeds increases (**Fig. 2c, Table S10**); this is because the more seeds there are on the plate lid, the more likely it is that seeds touch each other, leading to an increase in the false negative rate of prediction. Moreover, the detection of seeds could be disrupted by scratches or letters on the plate lids, and seeds outside the predefined circular search regions were not detected (purple arrowheads in **Fig. S4**). Thus, to obtain accurate counts based on segmentation, it is necessary to separate seeds and confine them to the center of the plate lid, which is time consuming and not amenable to high-throughput analysis.

Improved seed counting by an object detection method based on Faster R-CNN

Next, we evaluated the performance of an object detection approach using Faster R-CNN in seed counting. Since it is time-consuming to annotate a large number of seeds for model training, we adopted a two-step strategy. First, we split the 256 whole-plate images into 1024 quarter-plate images, and manually labeled a subset (180) of these quarter-plate images to speed up the training process. A total of 160 labeled quarter-plate images (*Training image set 1* in **Fig. 3a**) were used to build the models, and the remaining 20 images were set aside as the *validation image set* (**Fig. 3a**) to evaluate model performance. A model ($Model_{seed}$ 66) built with the optimal hyperparameter combination (scale-B, aspect ratio-A and 10,000 proposals, see **Methods S1**) was used to detect seeds

in the remaining 844 quarter-plate images to produce "*in silico*" seed annotations for the second-round modeling (**Fig. 3a,b**), resulting in 211 labeled whole-plate images.

A new model, $Model_{seed}$ 67, with the same parameters as $Model_{seed}$ 66, was built using 161 (*Training image set 2* in **Fig. 3b**) out of these 211 images. The remaining 50 labeled whole-plate images (*Test image set* in **Fig. 3b**) were used to evaluate the performance of $Model_{seed}$ 67, which had an improved average F1 of 0.992 (**Table S10**) compared with the F1 (~0.970) of $Model_{seed}$ 66 (**Fig. S2**). Note that the test set images were not used for training or validating $Model_{seed}$ 67; they were thus ideal for independently testing the model. In contrast to the segmentation method, $Model_{seed}$ 67 correctly predicted seeds even if they were in contact with each other (**Fig. 2b**), and the prediction accuracy was not influenced by the total seed number (r^2=0.9996, p=1.7e-83, **Fig. 2d**). The differences between true and predicted seed counts were close to zero, much smaller than those in segmentation-based analysis (**Fig. 2e**). Furthermore, $Model_{seed}$ 67 allowed the detection and counting of seeds in about 240 whole-plate images per hour using 1 GPU (Nvidia Tesla K80) with 4 GB of GPU memory in a UNIX cluster, or about 33 images per hour using a laptop with 16 GB of memory (i.e., ~800 seed images can be processed per day). These results suggest that our Faster R-CNN-based models provide highly accurate Arabidopsis seed counts and can be used for large-scale fitness studies.

Impact of seed density on the Faster R-CNN model

The number of seeds in an image has a detrimental effect on the performance of the segmentation method, but not on that of Faster R-CNN (**Fig. 2d**). To verify that the Faster R-CNN model performance was not affected by the seed density, we established the seed density index (SDI), which takes into account the differing densities across a single plate. First, a circle with a radius of 30 pixels (corresponding to 0.62 mm, approximate length of two seeds) was drawn from the center of a seed, then the number of seeds with central points located within the circle were calculated. Finally, the average number of seeds per circle in a whole-plate image was defined as the SDI (**Fig. 4a**).

We calculated the SDIs of the test set images (for examples see **Fig. S5**) and determined the Pearson's Correlation Coefficient (PCC) between SDI and the

performance of $Model_{seed}$ 67 on the test set images (**Fig. 4b**). The higher the seed density, the lower the model performance (PCC between SDI and F1 was -0.581, p=9.8e-06, **Fig. 4b**; for the correlation between SDI and other performance measures see **Fig. S6**). Nevertheless, the effect of seed density on the performance of $Model_{seed}$ 67 was small, as the F1 only dropped from 1.000 for an SDI of 1.157 to 0.971 for an SDI of 3.100 (**Fig. 4b, Table S10**). An F1 of 0.971 with a recall of 0.968 indicates that for an image with 1000 seeds, there would only be 32 false negatives (seeds not detected) and 25 false positives (seeds detected in an area with no seeds or a seed area counted more than once). Consistent with this, there was no significant correlation between the SDI and the difference between true and predicted seed counts (PCC=-0.206, p=0.15), in contrast to the significant negative correlation observed for the segmentation method (PCC=-0.886, p=1.2e-17, **Fig. 2f**). We also calculated SDIs for the predicted seed coordinates and found that the PCC value between true and prediction-based SDIs was 0.997 (p=1.5e-54; **Fig. 4c**), demonstrating that our Faster R-CNN model also predicts the locations of seeds very well.

Model improvement through data augmentation

Our goal is to provide a seed counting model that can be widely used by different researchers, who may have seed images with different properties. Thus, we investigated the utility of $Model_{seed}$ 67 using images with varying resolution, contrast, brightness, and blurriness (**Fig. 5a**). These modified seed images were created by modifying the properties of the test set images (**Fig. 3b**, for the image property settings see **Table S11**). In the modified test set, there were 1750 images: the original test set images (50) and modified images with 34 different attributes (34 × 50, light green box, **Fig. 3b**). A slight but significant decrease in F1 was observed when the brightness of the images was ≤ 0.60 (p=0.01, one-sided Wilcoxon signed-rank test) relative to the original images, while the F1 dropped dramatically when the relative brightness was ≥ 1.20 (p=6.4e-08, **Fig. 5b**). A significant decrease in F1 was also observed when the relative contrast of images (relative to the original image) was ≤ 0.50 (p=1.0e-07) or ≥ 1.75 (p=5.0e-4), the relative blurriness was ≥ 1.50 (p=6.7e-10), or the relative resolution was ≤ 0.50 (p=9.1e-10, **Fig.**

5b). These results suggest that although $Model_{seed}$ 67 is suitable for a range of image qualities, the seed detection accuracy will decrease dramatically when the image properties deviate from the training images beyond a certain point.

To improve the robustness of $Model_{seed}$ 67, we applied data augmentation, in which the size and properties of training datasets are increased so better prediction models can be built (Shorten & Khoshgoftaar, 2019). To accomplish this, we used 20 of the 161 training set 2 images to produce additional images with 21 different property settings (21 × 20, darker green box, **Fig. 3b,** for the image property settings see **Table S11**). These 420 additional images, together with the original 161 images, were used to build a new model, $Model_{seed}$ 68 (**Fig. 3b**), with the same hyperparameter settings as $Model_{seed}$ 67. $Model_{seed}$ 68 was then used to detect seeds in the modified test set images. Although there was a slight decrease in F1 when the relative blurriness was ≥ 3.00 (p=0.04, median F1 decrease=0.002) or when the relative resolution was ≤ 0.30 (p=0.02, median F1 decrease=0.003, **Fig. 5b**), $Model_{seed}$ 68 (blue, **Fig. 5b**) performed better than the non-augmented $Model_{seed}$ 67 (red, **Fig. 5b**) in all situations and thus, the augmented model is robust to different image properties.

Fruit counting using Faster R-CNN models

Compared with seed number, total fruit count is an even more frequently used proxy for fitness. Because dry Arabidopsis fruits shatter easily, it is not always possible to harvest all fruits produced by a single plant after seeds have matured, especially for plants growing in the field. In this case, the best method would be to count all fruits (including dehisced ones) and count seeds per fruit for a subset that haven't dehisced, and then calculate total seed number by multiplying the number of seeds per fruit by the total fruit number. Thus, to obtain more accurate estimates of seed production per plant, it is necessary to record the numbers of both intact and shattered fruits. With these considerations in mind, we developed Faster R-CNN models to count all fruits without harvesting the fruits first. When capturing the images for fruit counting, a pink background was used to maximize the contrast between the background and the dark, dry fruits and the pale replum of shattered fruits that remained after the valves fell from the

fruit (**Fig. 6a,d**). Because fruits in each image were less abundant and much larger compared with seeds, we manually labeled the fruits in 120 images.

Eighty, 20, and 20 images were randomly selected and used as training, validation, and test sets, respectively (**Fig. 6a**). Different combinations of hyperparameter values (**Table S8**) were evaluated and the resulting models ($Model_{fruit}$ 1–75, **Fig. 6a**) had similar performances with an average F1 of 0.925 (**Fig. S7**). Thus, to minimize the computational cost (lower scales or aspect ratios) while maximizing the number of fruits detected per plant (more proposals), the model built with $scale_{fruit}$-A, aspect $ratio_{fruit}$-A, and 500 proposals ($Model_{fruit}$ 21) was used. $Model_{fruit}$ 21 was applied to the test set images, resulting in an average F1 of 0.914 (**Table S12**). This F1 value translates into one false positive and 15 false negatives for an image with 100 fruits. Although the r^2 between true and predicted fruit counts was 0.980 (p=6.7e-17), the detection error increased with an increasing number of fruits in an image and the error was mostly due to undercounting or false negatives (**Fig. 6b,c**). The majority of the false negatives were unopened fruits that overlapped with the stem or with each other. One potential reason for the failure to detect these fruits is that they are similar to the stem in color and shape. Another reason may be the smaller number of labeled intact fruits (543) compared with the number of pale replums (2082) in our training images.

To assess the robustness of our model on images with different qualities, we applied $Model_{fruit}$ 21 on test set images with different image properties (**Fig. 6d,** modified test set, 700 images, for the image property settings see **Table S11**). Significant decreases in F1 were observed when the relative image brightness was ≤ 0.70 (p=0.04) or ≥ 1.40 (p=0.02), the relative contrast was ≤ 0.50 (p=0.02) or ≥ 1.50 (p=0.03), the relative blurriness was ≥ 2.0 (p=0.002), or the relative resolution was ≤ 0.6 (p=0.05) (**Fig. 6e**). By including images with different properties (**Table S11**) in the training set (1840 images), a new model, $Model_{fruit}$ 76, was established and applied to the modified test set. A significant but slight decrease in the resulting F1 values was only observed when the relative resolution was ≤ 0.3 (p=0.02, median F1 decrease=0.01) (**Fig. 6e**), indicating the robustness of $Model_{fruit}$ 76. Using this model 180 images could be processed per hour using a UNIX node with 1 GPU and 4 GB graphics memory, and 90 images per hour

could be processed using a laptop (1 CPU, 16 GB memory). Thus, our Faster R-CNN-based models can process over a thousand plant images per day.

Effects of loss of gene function revealed by measuring fitness traits

To evaluate the importance of fitness traits in investigating gene functions and the utility of our pipeline, the fruits and seeds produced by loss-of-function mutants of six pairs of duplicate genes (**Tables S3-S5**) were counted and compared with those of WT. Of these 12 mutants, three (*pap2*, *kin7.4*, and *hon5*) showed a significant difference in total seed count compared with the corresponding WT control (**Fig. 7 and Fig. S8, S9**). One of these genes, *PAP2*, modulates carbon metabolism; in addition, overexpression of *PAP2* resulted in earlier bolting and a higher seed yield than WT (Sun *et al*., 2012), which is consistent with the lower fitness that we observed for the *pap2* mutant (total seed counts, p=3.6e-03, Wilcoxon rank-sum test, **Fig. 7b**). However, when studying this same mutant, Sun *et al*. observed no significant differences in plant growth or seed yield relative to WT (Sun *et al.*, 2012).

One possible explanation for this discrepancy is the different fitness measures used by Sun *et al*.—seed weight per plant, seed weight per 100 seeds, and fruit number per plant—none of which were significantly different between *pap2* and WT (Sun *et al.*, 2012). To compare our fitness estimates more directly with those of Sun *et al*., we measured the same traits and found no significant difference in fruit number (p=0.15, **Fig. 7a**) or total seed weight per plant (p=0.40, **Fig. 7c**). However, the *pap2* mutant did have a higher weight per 100 seeds than the WT (p=3.8e-08, **Fig. 7d**). This could potentially indicate differences in viability because larger seeds have more resources for germination and early seedling growth (Sundaresan, 2005), but we observed no difference in germination rate between WT and *pap2* (**Table S4**), suggesting that there is no difference in seed viability. Taken together, our findings suggest that seed number is a better measure for revealing fitness effects of loss of *PAP2* function. However, we cannot rule out the possibility that we observed these effects because our experimental conditions were more stressful (i.e., nutrient limiting) than those in Sun *et al*. (2012).

For *GH1-HMGA2/HON5*, which encodes a high-mobility group protein (Kotliński *et al.*, 2017), and *KIN7.4*, which belongs to the kinesin motor family, members of which are involved in microtubule-based movement (Moschou *et al.*, 2016),there were significant differences in both fruit numbers (p=0.04 for *hon5 and* p=5.0e-04 for *kin7.4* , **Fig. 7e,g**) and seed numbers (p=5.8e-03 for *hon5 and* p=3.0e-05 for *kin7.4* , **Fig. 7f,h**) between the mutants and WT. No functions have been reported for *KIN7.4*. *HON5* was previously shown to regulate the transition to flowering along with *HON4* by repressing *FLC* expression, but no effects on fitness were reported (Zhao *et al.*, 2021). Loss of function of *HON4* was previously reported to cause sterility (Charbonnel *et al.*, 2018), but neither we nor Zhao *et al*. (2021) observed this phenotype when using a different mutant with an insertion in a similar location (intron 2), suggesting that the sterility phenotype of the *hon4* mutant may be dependent on environmental conditions.

Discussion

Fitness is one of the best measures of gene functionality because it reflects the ability of a plant to survive and reproduce given all the phenotypic effects of the mutation over the lifetime of the individual. For self-pollinating species such as Arabidopsis, fitness is better assessed by counting the numbers of seeds than fruits, as they more directly reflect the number of offspring and reproductive success. Because of the lack of an effective tool enabling high throughput counting of small seeds *en masse*, seed counts are often estimated indirectly, for example by dividing the total seed weight per plant by the estimated individual seed weight (Cvetkovic *et al.*, 2017), or multiplying the fruit count by the average fruit length (Kerwin *et al.*, 2015; Taylor *et al.*, 2019). However, these approaches may not yield accurate estimates of seed production because of the imperfect correlation between seed number and fruit length (Roux *et al*., 2004). Here, we established a model employing a deep learning approach, Faster R-CNN, to count Arabidopsis seeds—one of the smallest objects analyzed using machine vision to date—with a near perfect accuracy (F1=0.992) using images with multiple different properties or qualities.

Our model outperforms the Mask R-CNN approaches in Toda *et al.*, (2020) (F1 of about 0.95), where the detected objects were much larger than Arabidopsis seeds. Mask R-CNN is built on top of Faster R-CNN so the differences in performance likely are not due to differences in algorithms. The better performance of our model is likely because our training seed images are more representative of the diversity in seed sizes and shapes than the repetitive cropped images used by Toda *et al*. The Faster R-CNN-based predictions greatly outperform those of the segmentation method implemented in ImageJ, a well-known platform with macros/modules for segmentation and morphology extraction (Schneider *et al.*, 2012; Cervantes *et al.*, 2016; Vasseur *et al.*, 2018). In addition, object detection based on Faster R-CNN is less time consuming than segmentation using ImageJ because seeds can be accurately detected without first being separated or confined to predefined regions.

One of the challenges when using deep learning approaches is the requirement for a large number of labeled data (in our case, labeled seeds). To overcome this, we adopted a two-step modeling strategy to reduce the labor needed for seed annotations. In step 1, we split the images and used a subset of the split images to build a preliminary model ($F1<0.975$) and applied it to the remaining images. While the predictions were not perfect, this step drastically reduced the manual annotations needed because we only needed to correct mis-predictions to boost our seed labels by ~5 fold (29,360 labels in the first-round, 138,929 labels in the second-round). Using this much larger set of seed labels, new models were built (step 2) that had improved model performance ($F1=0.992$), indicating the effectiveness of our strategy.

The Faster R-CNN approach also shows promise in fruit detection and counting ($r^2=0.98$, slope=0.79). The performance of our fruit counting model was better than that of another recently published CNN-based approach, DeepPod ($r^2=0.90$, slope ~0.70, Hamidinekoo *et al.*, 2020). In that paper, the task (i.e., fruit detection) was first divided into four classification tasks: the detection of the tip, body, and base of the fruits and the detection of the stem. The separately detected parts were then joined together as a whole fruit. As the authors noted, this post-processing step affected the final fruit detection performance. In our study, the fruits were labeled and detected as whole objects, thus avoiding the need for post-processing. In addition, different from Hamidinekoo *et al.*

(2020), where most fruits and the stems in the images were fresh and green, fruits in our study were dry and light brown to gray, or were shattered with only the pale replum remaining. Thus, our fruit counting approach is expected to be applicable to a wider range of Arabidopsis fruit developmental stages. This is especially important when plants must be grown to maturity, and seed counts are estimated by multiplying the average number of seeds per intact fruit by the total number of fruits (intact and dehisced) (Conner & Rush, 1997).

Nevertheless, our fruit counting models did not perform as well as our seed counting models and a published ImageJ-based segmentation and skeletonization approach (r^2=0.91, slope= ~1; Vasseur *et al.*, 2018), which may be due to the much fewer labeled fruits than labeled seeds (there were about 52 times more labeled seeds than fruits). Thus, the performance of the fruit counting model is expected to be improved when more fruit labels are included to train the model. In addition, one notable drawback of our approach is the undercounting at higher fruit numbers; this was mainly due to overlap between intact fruits and between intact fruits and stems. To remedy this, one approach is to rearrange the inflorescences before capturing the images to keep fruits from overlapping with each other and with stems. Another potential approach, which is an important future direction, is to analyze multiple images (or frames of a movie) taken at different angles or to examine the 3D reconstruction of the inflorescence. In addition, there have been substantial advances in object detection algorithms in terms of performance and processing speed. New initial models that can be retrained (e.g., Inception v.3 and v.4) have also been developed (we used Inception v.2). Although we explored some of these algorithms and initial models (see **Methods S1**), we did not optimize them because of the significant computational complexity in just optimizing Faster R-CNN/Inception v.2 for fitness traits. Thus, in future studies, these algorithms and initial models should be more thoroughly explored to further improve fitness trait phenotyping.

We should emphasize that the picture of seeds or fruits are taken for record keeping and documentation purposes regardless of whether a machine-vision-based approach or manual counting is used. After the picture is available, it takes our Faster R-CNN-based models about 109 and 40 seconds to provide counts for a seed and fruit picture, respectively. In contrast, manual counting takes us about 50 seconds per 100 seeds and 40

seconds per 100 fruits. Thus, as the seed and fruit number increases, our Faster R-CNN-based models have an even bigger advantage over manual counting.

By examining fitness traits, especially seed counts, we were able to observe phenotypic changes in loss-of-function mutants that were previously not detectable (*pap2*, Sun *et al.*, 2012) or not reported (*kin7.4* and *hon5*). In our relatively small sample of twelve mutants, effects on fitness were observed for three (25%). A similar percentage of lines with lower fitness than WT was reported by Rutter *et al.*, (2017), who investigated the fitness effects of Arabidopsis T-DNA insertion lines using fruit number as a measure. They also found that a sizable percentage of lines had increased fitness compared with WT (12%), leading them to conclude that genetic redundancy is not common. We found that fruit counts could reveal fitness effects for two of three genes, indicating that seed counts are a better measure of fitness in some cases, such as when a genotype produces more fruits with fewer seeds per fruit. We are currently measuring both seed and fruit counts for a large number (>400) of mutants, which will allow us to obtain a more complete picture of the relative importance of fruit and seed counts for assessing fitness.

The seed counting pipeline that we established does not measure seed size, which is an agriculturally important trait associated with yield and seed viability (Sundaresan, 2005). By measuring seed weights, we found that *pap2* produces larger seeds than WT. Although we observed no clear difference in viability between them, seed size is a useful distinguishing characteristic between these genotypes. It might also provide insight into the underlying biology. For example, one possible reason for the increased seed size in the mutant is a lower fertilization rate, which would lead to fewer seeds and less restriction on seed growth (Herridge *et al.*, 2011; Fatihi *et al.*, 2013). Because measuring seed weights is time consuming, a focus of our future work will be to adapt our pipeline to include approaches to measure seed size and number simultaneously.

Taken together, our results illustrate the importance of fitness traits in the study of gene functions and show that Faster R-CNN-based models, which can almost perfectly detect and count Arabidopsis seeds and also detect fruits with high accuracy, are valuable tools in large-scale studies of plant fitness. In the future we will use these tools to

548 measure the fitness traits of a larger number of mutants to obtain a more complete picture
549 of the effects of loss of gene function on fitness.
550
551 ## Figure legends

552 **Fig. 1.** Workflow and performance for seed counting with a segmentation method using
553 ImageJ when seeds were deliberately separated. (**a**) Workflow. Seeds from 12 different
554 plants were scattered and manually separated from each other on the lids of 12 petri
555 plates, which were placed in a template and scanned. Twelve search areas, each with a
556 diameter of 60 mm (yellow circles), were predefined. A threshold was applied by
557 selecting pixels with intensities between 50 and 140 to separate the seed areas (red) from
558 the background. Then pixels were converted to real-world distance units in mm. The
559 "Analyze Particles" tool was used to detect and count the seeds. (**b**) An example of an
560 image with detected seeds (left) and an enlarged image showing the seeds (right). Red
561 region with number: individual detected seed area. (**c**) Correlation between true and
562 predicted seed counts using the segmentation method when seeds were deliberately
563 separated.
564
565 **Fig. 2.** Comparison between the performances of the segmentation and Faster R-CNN-
566 based seed counting methods for test set images of seeds that were not deliberately
567 separated. (**a, b**) The same seed scan image analyzed by the segmentation method using
568 ImageJ (**a**) and by Faster R-CNN (**b**). Three different regions of the plate lid with
569 different densities are outlined. Region 1 has low seed density, region 2 has moderate
570 density, and region 3 has a high density. In (**a**) the red colored regions represent the
571 segmented areas identified by the segmentation method; seeds outlined in yellow and
572 assigned numeric IDs were counted. In (**b**) the blue rectangles represent seeds detected by
573 Faster R-CNN. (**c,d**) Correlation between true and predicted seed numbers from
574 segmentation method (**c**) and Faster R-CNN (**d**) analysis of the test set. (**e**) Distribution of
575 differences between true and predicted seed numbers. Red lines: the segmentation
576 method using ImageJ; blue lines: Faster R-CNN. (**f**) Correlation between seed density
577 index (SDI) and difference between true and predicted seed counts. Each dot in (**c,d,f**)

corresponds to one of the 50 test set images. The red line in (**c**) is the regression line obtained using the loess method. The blue lines in (**d,f**) are fitted regression lines for Faster R-CNN predictions. The red line in (**f**) is the fitted linear regression line for the segmentation method-based predictions. PCC: Pearson correlation coefficient.

Fig. 3. Workflow for building Faster R-CNN-based seed counting models. (**a**) First-round modeling for enriching annotated seed labels. Each of the 256 whole-plate images was split into four quarter-plate images. Among the 1024 quarter-plate images, 180 were used in first-round modeling, and the remainder (844) were used in second-round modeling described in (**b**). Seeds in the 180 quarter-plate images were manually annotated, and then these annotated images were further split into training set 1 (160) and a validation set (20) to train and evaluate models, respectively. Sixty-three combinations of three hyperparameters (i.e., 3 proposal numbers × 3 scales [A, B, and C] × 7 aspect ratios [AR-A through G]; for scale and aspect ratio values see **Table S6**) were used to build 63 models. The optimal scale (B) and aspect ratio (AR-A) were selected based on the model performance on validation set images (**Fig. S2**). An additional three models ($Model_{seed}$ 64–66) were built using scale B, AR-A, and three larger proposal values, and the final best model, $Model_{seed}$ 66, with 10,000 proposals, was applied to the 844 quarter-plate images reserved for second-round modeling to generate *in silico* seed annotations. (**b**) Second-round modeling. The 844 quarter-plate images with seed predictions from $Model_{seed}$ 66 were rejoined together to reconstruct 211 whole-plate images with *in silico* seed annotations, which were then manually curated and used as ground truth seed annotations. $Model_{seed}$ 67 was built using 161 (training set 2) out of the 211 annotated images with the same hyperparameters used in $Model_{seed}$ 66, and was evaluated using the test set (50 independent images not used for modeling) and the modified test set (i.e., the 50 independent test set images plus 1,700 images modified from the test set images that had different image properties [blurriness, brightness, contrast, and resolution values]). For data augmentation, the image properties of 20 images from training set 2 were modified, and the resulting 420 images were combined with training set 2 (161 images), resulting in 581 images (modified training set 2), which were used to build $Model_{seed}$ 68. The modified test set was used to evaluate the performance of $Model_{seed}$ 68.

Fig. 4. Effect of seed density on the performance of the Faster R-CNN models. (**a**) Examples with different seed density index (SDI) values. The radius of each circle is 30 pixels (0.62 mm). (**b,c**) Relationship between SDI and model performance (**b**) and between the true SDI and SDI based on prediction (**c**) for test images. Each dot corresponds to one of 50 test set images. Blue lines are the fitted linear regression lines. F1: F1 value at 0.5 IoU (Intersection over Union). PCC: Pearson correlation coefficient.

Fig. 5. Improvement of model robustness using training images with different properties. (**a**) Examples of seed images with different relative brightness, contrast, blurriness, and resolution values that were derived from the same original image. (**b**) Model performance for $Model_{seed}$ 67 and $Model_{seed}$ 68 on the modified test set (**Fig. 2b**). F1: F1 value at 0.5 IoU (Intersection over Union); red boxplot: $Model_{seed}$ 67; blue boxplot: $Model_{seed}$ 68; horizontal line in the box: median value; box range: interquartile range (IQR), i.e., 25^{th} (Q1) to 75^{th} percentile (Q3); whisker below box: Q1 – 1.5*IQR to Q1; whisker above box: Q3 to Q3 + 1.5*IQR.

Fig. 6. Fruit counting using Faster R-CNN models. **(a**) Fruit counting workflow. (**b**) Relationship between true and predicted fruit numbers. (**c**) Relationship between fruit number in an image and the model performance. PCC: Pearson correlation coefficient. (**d**) Examples of the same fruit image with different relative brightness, contrast, blurriness, and resolution values. (**e**) Model performance for $Model_{fruit}$ 21 and $Model_{fruit}$ 76 on test images with different properties. F1: F1 at 0.5 IoU (Intersection over Union); red boxplot: $Model_{fruit}$ 21; blue boxplot: $Model_{fruit}$ 76; horizontal line in the box: median value; box range: interquartile range (IQR), i.e., 25^{th} (Q1) to 75^{th} percentile (Q3); whisker below box: Q1 – 1.5*IQR to Q1; whisker above box: Q3 to Q3 + 1.5*IQR.

Fig. 7. Fitness measurements for three mutants. (**a-d**) Fruit counts per plant (**a**), seed counts per plant (**b**), seed weight per plant (**c**), and weight per 100 seeds (**d**) for the T-DNA insertion mutant of *PURPLE ACID PHOSPHATASE 2* (*pap2*) and wild type (WT). (**e-f**) Fruit (**e**) and seed (**f**) counts per plant, for the T-DNA insertion mutant of *HIGH*

MOBILITY GROUP A5 (*hon5*) and WT. (**g-h**) Fruit (**g**) and seed (**h**) counts per plant for the T-DNA insertion mutant of *KINESIN 7.4* (*kin7.4*) and WT. Sample sizes are shown in parentheses on the x axis. *p*-values are from Wilcoxon signed-rank tests. Horizontal line in the box: median value; box range: interquartile range (IQR), i.e., 25th (Q1) to 75th percentile (Q3); whisker below box: Q1 – 1.5*IQR to Q1; whisker above box: Q3 to Q3 + 1.5*IQR; violin plot: distribution of datapoint values; dot: datapoint from an individual plant; yellow: loss-of-function mutant; cyan: WT.

Acknowledgments

We thank Christina B. Azodi, Bethany M. Moore, Siobhan Cusack, and Liang Xu for help in manual seed annotation, and Ally Schumacher for providing the photo of the template. We thank Dirk Colbry for helpful discussions. This work was supported by the U.S. Department of Energy Great Lakes Bioenergy Research Center (BER DE-SC0018409) and the National Science Foundation (DEB-1655386 to JKC and SHS; DGE-1828149 to SHS; IOS-2107215 to MDL and SHS; and DEB-1655630 to PJK).

Author contributions

PW, FM, JKC, PJK, MDL, and SHS conceived and designed the study. PW, FM, PD, SH, NLP, EV, EW, JKC, PJK, and MDL performed data collection and analysis. PW, FM, MDL, and SHS wrote the manuscript. All authors read and approved the final manuscript.

Data availability

All the scripts used in this study and the final seed and fruit counting models are available on Github at: https://github.com/ShiuLab/Manuscript_Code/tree/master/2022_Arabidopsis_seed_and_fruit_count

References

Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M, et al. 2016. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *arXiv*:1603.04467 [cs].

Afonso M, Fonteijn H, Fiorentin FS, Lensink D, Mooij M, Faber N, Polder G, Wehrens R. 2020. Tomato Fruit Detection and Counting in Greenhouses Using Deep Learning. *Frontiers in Plant Science* **11**: 571299.

Bouché N, Bouchez D. 2001. Arabidopsis gene knockout: phenotypes wanted. *Current Opinion in Plant Biology* **4**: 111–117.

Boyes DC, Zayed AM, Ascenzi R, McCaskill AJ, Hoffman NE, Davis KR, Görlach J. 2001. Growth Stage–Based Phenotypic Analysis of Arabidopsis: A Model for High Throughput Functional Genomics in Plants. *The Plant Cell* **13**: 1499–1510.

Busoms S, Teres J, Huang X-Y, Bomblies K, Danku J, Douglas A, Weigel D, Poschenrieder C, Salt DE. 2015. Salinity Is an Agent of Divergent Selection Driving Local Adaptation of Arabidopsis to Coastal Habitats. *Plant Physiology* **168**: 915–929.

Cao C, Wang B, Zhang W, Zeng X, Yan X, Feng Z, Liu Y, Wu Z. 2019. An Improved Faster R-CNN for Small Object Detection. *IEEE Access* **7**: 106838–106846.

Cervantes E, Martín JJ, Saadaoui E. 2016. Updated Methods for Seed Shape Analysis. *Scientifica* **2016**: 1–10.

Charbonnel C, Rymarenko O, Da Ines O, Benyahya F, White CI, Butter F, Amiard S. 2018. The Linker Histone GH1-HMGA1 Is Involved in Telomere Stability and DNA Damage Repair. *Plant Physiology* **177**: 311–327.

Conner JK, Rush S. 1997. Measurements of selection on floral traits in black mustard, *Brassica Nigra*. *Journal of Evolutionary Biology* **10**: 327.

Cvetkovic J, Müller K, Baier M. 2017. The effect of cold priming on the fitness of Arabidopsis thaliana accessions under natural and controlled conditions. *Scientific Reports* **7**: 44055.

Fatihi A, Zbierzak AM, Dörmann P. 2013. Alterations in Seed Development Gene Expression Affect Size and Oil Content of Arabidopsis Seeds. *Plant Physiology* **163**: 973–985.

Gnan S, Priest A, Kover PX. **2014**. The Genetic Basis of Natural Variation in Seed Size and Seed Number and Their Trade-Off Using *Arabidopsis thaliana* MAGIC Lines. *Genetics* **198**: 1751–1758.

Halcro K, McNabb K, Lockinger A, Socquet-Juglard D, Bett KE, Noble SD. **2020**. The BELT and phenoSEED platforms: shape and colour phenotyping of seed samples. *Plant Methods* **16**: 49.

Hamidinekoo A, Garzón-Martínez GA, Ghahremani M, Corke FMK, Zwiggelaar R, Doonan JH, Lu C. **2020**. DeepPod: a convolutional neural network based quantification of fruit number in Arabidopsis. *GigaScience* **9**: giaa012.

Hasan MM, Chopin JP, Laga H, Miklavcic SJ. **2018**. Detection and analysis of wheat spikes using Convolutional Neural Networks. *Plant Methods* **14**: 100.

Herridge RP, Day RC, Baldwin S, Macknight RC. **2011**. Rapid analysis of seed size in Arabidopsis for mutant and QTL discovery. *Plant Methods* **7**: 3.

Hirsch RE, Lewis BD, Spalding EP, Sussman MR. **1998**. A Role for the AKT1 Potassium Channel in Plant Nutrition. *Science* **280**: 918–921.

Huang J, Rathod V, Sun C, Zhu M, Korattikara A, Fathi A, Fischer I, Wojna Z, Song Y, Guadarrama S, *et al.*2017. Speed/Accuracy Trade-Offs for Modern Convolutional Object Detectors. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI: IEEE, 3296–3297.

Jahnke S, Roussel J, Hombach T, Kochs J, Fischbach A, Huber G, Scharr H. **2016**. phenoSeeder - A Robot System for Automated Handling and Phenotyping of Individual Seeds. *Plant Physiology* **172**: 1358–1370.

Kerwin R, Feusier J, Corwin J, Rubin M, Lin C, Muok A, Larson B, Li B, Joseph B, Francisco M, *et al.*2015. Natural genetic variation in Arabidopsis thaliana defense metabolism genes modulates field fitness. *eLife* **4**: e05604.

Kotliński M, Knizewski L, Muszewska A, Rutowicz K, Lirski M, Schmidt A, Baroux C, Ginalski K, Jerzmanowski A. **2017**. Phylogeny-Based Systematization of Arabidopsis Proteins with Histone H1 Globular Domain. *Plant Physiology* **174**: 27–34.

Mauricio R, Rausher MD. **1997**. Experimental manipulation of putative selective agents provides evidence for the role of natural enemies in the evolution of plant defense.

Evolution **51**: 1435–1444.

McDonald JF. **1983**. The Molecular Basis of Adaptation: A Critical Review of Relevant Ideas and Observations. *Annual Review of Ecology and Systematics* **14**: 77–102.

Meissner RC, Jin H, Cominelli E, Denekamp M, Fuertes A, Greco R, Kranz HD, Penfield S, Petroni K, Urzainqui A, *et al.*1999. Function Search in a Large Transcription Factor Gene Family in Arabidopsis: Assessing the Potential of Reverse Genetics to Identify Insertional Mutations in R2R3 *MYB* Genes. *The Plant Cell* **11**: 1827–1840.

Mochida K, Koda S, Inoue K, Hirayama T, Tanaka S, Nishii R, Melgani F. **2019**. Computer vision-based phenotyping for improvement of plant productivity: a machine learning perspective. *GigaScience* **8**: giy153.

Moore CR, Johnson LS, Kwak I-Y, Livny M, Broman KW, Spalding EP. **2013**. High-Throughput Computer Vision Introduces the Time Axis to a Quantitative Trait Map of a Plant Growth Response. *Genetics* **195**: 1077–1086.

Morales A, Teapal J, Ammerlaan JMH, Yin X, Evers JB, Anten NPR, Sasidharan R, van Zanten M. **2020**. A high throughput method for quantifying number and size distribution of Arabidopsis seeds using large particle flow cytometry. *Plant Methods* **16**: 27.

Moschou PN, Gutierrez-Beltran E, Bozhkov PV, Smertenko A. **2016**. Separase Promotes Microtubule Polymerization by Activating CENP-E-Related Kinesin Kin7. *Developmental Cell* **37**: 350–361.

Penaloza P, Ramirez-Rosales G, B. McDonald M, A. Bennett M. **2005**. Lettuce (Lactuca sativa L.) seed quality evaluation using seed physical attributes, saturated salt accelerated aging and the seed vigour imaging system. *Electronic Journal of Biotechnology* **8**: 299–307.

Ren S, He K, Girshick R, Sun J. **2017**. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39**: 1137–1149.

Roux F, Gasquez J, Reboud X. **2004**. The Dominance of the Herbicide Resistance Cost in Several *Arabidopsis thaliana* Mutant Lines. *Genetics* **166**: 449–460.

Rutter MT, Wieckowski YM, Murren CJ, Strand AE. **2017**. Fitness effects of

mutation: testing genetic redundancy in *Arabidopsis thaliana*. *Journal of Evolutionary Biology* **30**: 1124–1135.

Schneider CA, Rasband WS, Eliceiri KW. **2012**. NIH Image to ImageJ: 25 years of image analysis. *Nature Methods* **9**: 671–675.

Shorten C, Khoshgoftaar TM. **2019**. A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data* **6**: 60.

Sun F, Suen PK, Zhang Y, Liang C, Carrie C, Whelan J, Ward JL, Hawkins ND, Jiang L, Lim BL. **2012**. A dual-targeted purple acid phosphatase in *Arabidopsis thaliana* moderates carbon metabolism and its overexpression leads to faster plant growth and higher seed yield. *New Phytologist* **194**: 206–219.

Sundaresan V. **2005**. Control of seed size in plants. *Proceedings of the National Academy of Sciences* **102**: 17887–17888.

Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. 2016. Rethinking the inception architecture for computer vision. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*:2818-2826. https://doi.org/10.1109/Cvpr.2016.308.

Tanabata T, Shibaya T, Hori K, Ebana K, Yano M. **2012**. SmartGrain: high-throughput phenotyping software for measuring seed shape through image analysis. *Plant Physiology* **160**: 1871–1880.

Taylor MA, Wilczek AM, Roe JL, Welch SM, Runcie DE, Cooper MD, Schmitt J. **2019**. Large-effect flowering time mutations reveal conditionally adaptive paths through fitness landscapes in *Arabidopsis thaliana*. *Proceedings of the National Academy of Sciences* **116**: 17890–17899.

Thomson CE, Hadfield JD. **2017**. Measuring selection when parents and offspring interact (J Stinchcombe, Ed.). *Methods in Ecology and Evolution* **8**: 678–687.

Toda Y, Okura F, Ito J, Okada S, Kinoshita T, Tsuji H, Saisho D. **2020**. Training instance segmentation neural network with synthetic datasets for crop seed phenotyping. *Communications Biology* **3**: 173.

Vasseur F, Bresson J, Wang G, Schwab R, Weigel D. **2018**. Image-based methods for phenotyping growth dynamics and fitness components in Arabidopsis thaliana. *Plant Methods* **14**: 63.

Zhao B, Xi Y, Kim J, Sung S. **2021**. Chromatin architectural proteins regulate flowering time by precluding gene looping. *Science Advances* **7**: eabg3097.

Supporting information

Fig. S1. The architecture of Faster R-CNN
Fig. S2. Hyperparameter tuning for seed counting models
Fig. S3. Computational efficiency of seed counting models
Fig. S4. Example false negatives from the segmentation method using ImageJ and Faster R-CNN models
Fig. S5. Example images with different SDI values
Fig. S6. Effect of seed density on the performance of the Faster R-CNN models using different measures of performance
Fig. S7. Hyperparameter tuning for fruit counting models
Fig. S8 Fitness measurements for T-DNA insertion mutants of 12 genes
Fig. S9 The proportion of fruits produced by twelve mutants that are shattered or green
Table S1. Lines used for training seed counting models
Table S2. Lines used for training fruit counting models
Table S3. Lines used for analysis of fitness
Table S4. Fitness data for *pap2*, *pap9*, *hon4*, *hon5*, *eif4b1*, and *eif4b2*
Table S5. Fitness data for *aprl5*, *aprl7*, *pfa-dsp3*, *pfa-dsp5*, *kin7.2*, and *kin7.4*
Table S6. Hyperparameter space for seed counting
Table S7. Seed counts in 20 quarter-plate images in the validation set
Table S8. Hyperparameter space for fruit (silique) counting
Table S9. Seed counting with the segmentation method using ImageJ
Table S10. Seed counting for 50 test set seed images using $Model_{seed}$ 67
Table S11. Image property settings for $Model_{seed}$ 68 and $Model_{fruit}$ 76
Table S12. Fruit counting for 20 test fruit images using $Model_{fruit}$ 21
Methods S1. Plant growth conditions, seed image processing, and seed counting.