Detection in agricultural contexts: Are we close to human level?

Anonymous CVPPP submission

Paper ID 29

Abstract. We consider detection accuracy in agricultural contexts. Five challenging datasets were collected and benchmarked, with three recent networks tested. Based on an initial analysis showing the importance of image resolution, models were trained and tested with a multiple-resolution procedure. Detection results were compared to human performance, judged based on the consistency of multiple annotators. A quantitative analysis was made highlighting the role of object scale and occlusion as detection failure causes. Finally, novel detection accuracy metrics were suggested based on the needs of agriculture tasks, and used in detector performance evaluation.

Keywords: Detection, Precision agriculture, Human performance

1 Introduction

Object detection in the agriculture environment is important for a variety of agricultural tasks and applications such as robotic manipulation, counting, and fine phenotyping. Robotic manipulation tasks as fruit [20] and vegetable [25] harvesting were recognized as an important task to automate more than 50 years ago [21]. Other robotic tasks requiring a detection module include plant spraying [3] and detection and handling of pests and diseases [7]. Counting tasks are common for the purpose of yield estimation [16, 26], or blooming intensity estimation [6], and at least in some approaches require explicit object detection. Fine phenotyping tasks involve examining an object's traits and features to evaluate a plant's growth, resistance, physiology condition, or any other observable parameter [4]. For example, in [1, 24] various length or height parameters of plant parts were estimated. A successful object detector is crucial for achieving practical performance in each of the above tasks.

Detecting objects in field or orchard conditions is not an easy task. In 2001 it was recognized by Li et al. [12] that improvements in detection and localization of objects are the main obstacles preventing harvesting robots from reaching human capabilities. In recent years, Convolutional Neural Networks (CNNs) based detectors dramatically improved, bridging some of the gap between human and machine performance. CNNs based detectors can be divided into two natural groups - single stage and two stage detectors. Single stage detectors, such as YOLO [17], RetinaNet [14], and EfficientDet [23], consider hundreds of thousands of possible object locations in the image, and classify them in a single

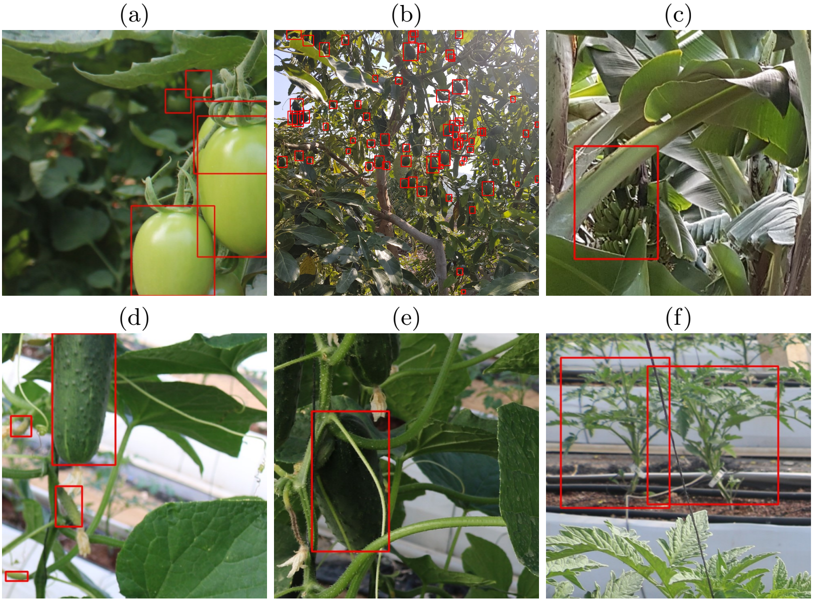


Fig. 1. Challenges for visual detection in the agricultural context. (a) severe occlusion and scale variations (b) dozens of avocado objects in a single image (c) severe occlusion (d) scale variation (e) poor illumination (f) challenging discrimination. Image (b) shows a full image, and the others are sub-images showing the difficulties

unified network. Two stage detectors, such as Faster-RCNN [18] or Mask R-CNN [9], start by generating a smaller set of object candidates (a few hundreds or thousands), then classify and refine them in a second network.

Detection in the field context is different in some characteristics from traditional detection benchmarks [5, 15] and in some respects more challenging. First, field images may contain dozens of objects, with high scale variance. Naturally both near and far objects are captured, and objects in many octaves often exist in a single image. Second, in many cases, such as apple flowers or tomatoes, the objects of interest grow in clusters. Hence occlusion is very common, with many objects suffering from high occlusion degrees. Third, the objects of interest often have a challenging shape with similarity to background structures. Tomatoes and avocados for example, are simple and round without discriminative details, and can often be confused with round leafs in the foliage. Cucumbers are green and stick-like, with high similarity to some branches and stalks. Tomato whole plants, on the other hand, are non-convex and skeletal. Finally, there are challenges introduced by the outdoors illumination conditions, including coping with severe cast shadows and required invariance to capturing hour. Some of these difficulties can be observed in figure 1.

With the rapid advance of detection networks, several questions arise with respect to the agricultural context: What are the better network architectures and training procedures for the agricultural domains? With the best models, is fruit and plant detection now approaching human level? If there is still a gap, can we characterize the main reasons for detection errors and quantify them? finally, assuming human level has not been reached yet: is the accuracy of current detectors satisfactory for practical applications? Which measurements may help us in answering this question? In this paper we try to make some progress regarding these issues.

In order to characterize detection performance we collected images from five different agricultural contexts, in which the tasks are detection of tomatoes, cucumbers, avocados, banana bunches, and whole tomato plants. These represent a diverse set of challenges related to non-discriminative shape (cucumbers), non-convex shape (whole plants), natural clustering and occlusion (tomatoes). Datasets were annotated with strict annotation, aiming to include all objects visible by a human (including small and highly occluded ones). To obtain good detectors, we have experimented with two main dimensions. First, we tested three leading network architectures: the two staged Mask R-CNN [9], the single-stage RetinaNet [14], and the recent EfficientDet [23]. More importantly, we conducted an analysis showing that accuracy is highly affected by object scales and processing resolutions. In light of that realization we have experimented with training and inference procedures involving multiple image resolutions.

The best detectors obtained were analysed in two informative respects: first, accuracy was compared to estimated human accuracy (on 4 of the 5 datasets where the required annotation existed). To estimate human accuracy, annotations of the same dataset made by 2 or 3 different annotators were used. Human performance was estimated based on the consistency between annotators, with one annotator operating in the 'predictor' role and the other as the 'ground truth'. A second analysis was pursued to quantify the role of object scale and occlusion in detection difficulty. To this end, additional occlusion annotation was added to the most challenging dataset in this respect, the tomato dataset. Detection accuracy was then measured for subsets of objects characterized by their size and occlusion level, and compared to the corresponding human performance.

The suitability of a certain detector to a certain application is clearly an application-dependent question, which cannot be answered here. However, we claim that in order to answer such questions, detector accuracy should be characterized with a richer set of measurements than the commonly used Average Precision (AP) or F_1 statistic. In robotic interaction, for example, object localization accuracy is most important. For phenotyping one usually does not have to detect all the objects [24], but measurements should not be made on non-objects. Hence recall at high precision levels is the most relevant. For counting, we claim that counting error for a certain confidence threshold is the suitable performance indicator. We suggest a set of accuracy measurements tailored to agriculture applications and measure the best detectors trained.

Our contribution in this work is four-fold. We report a benchmark of several recent detector architectures on a diverse dataset of five agricultural detection tasks. We analyze the effect of object scale on detection accuracy and show the importance of multiple resolution network processing for tasks containing a wide scale range. We analyze the performance of the best detectors with respect to error source and with comparison to human level performance. Finally, we suggest a set of detection accuracy measurements more tailored to common agricultural tasks, and use these to measure the best detectors trained.

2 Related Work

Object detection in agriculture is an extensively studied subject, with the outdoor field environment presenting unique challenges. In [8] variable lighting condition, occlusions, and fruit or flower clustering were mentioned as the main difficulties. Other works [2, 6] have acknowledged and faced the challenges of having many objects with small scale. Most early published work was based on explicit formation of color, texture or geometric features enabling detection of the target objects. The review [8] published in 2015, provides a good overview of these techniques. With the advance of CNN models in recent years, they were found a good fit to cope with the challenges, and avoid the manual feature construction. Given enough data, deep networks learn a good representation including discriminating features, which enable detection of target objects with accuracy superior to previous methods. A review of deep learning techniques in agriculture, including some examples of successful detection applications is presented in [11].

While there are numerous studies that use a deep learning based detector in agricultural tasks, we focus here on the benchmarks [19, 27, 2], which are the most relevant to our work. Sa and his collaborators [19] use Faster R-CNN [18] to detect sweet peppers and rock-melons by combining RGB and Near-IR information. Specifically, the two modalities were combined by adding the NIR map to the network input, and this addition is shown to contribute to accuracy (rising the F_1 score from 0.813 to 0.838). The work shows the generality of the approach by considering 7 different fruit kinds, and the F_1 results obtained are very good. However, the datasets used are significantly easier than in our work. Images are mostly taken in plantation conditions, with small distance between the camera and the relevant fruits. Hence typically an image includes less than 5 fruits (rarely more than 10), and these are usually big and clearly visible. In contrast, the images used in our experiments often contain many dozens of objects, and with significant scale variation including many small and far objects. In [27] a large dataset, containing $49,000$ annotated objects from 31 classes was collected and benchmarked for detection and classification with deep networks. However, this data is even more extreme than the datasets of [19], with most images containing a single large object of interest.

Bargoti and Underwood [2] used Faster R-CNN to detect mangoes, apples, and almonds. Their dataset is similar to ours with respect to the image size and the number of objects in each frame. To resolve scale and number of objects issues they also use a tiling system - image was divided into tiles of 500×500 pixels with 50 pixels overlap. While this work is the most similar to ours, our benchmark

Table 1. Data set sizes and partitions shown as (# images, #objects),image sizes, and object size statistics. Object size is in pixels, defined by $\sqrt{width \times height}$. The table shows mean object size and $std(log(size))$ with log base 2 in parenthesis

Crop	Train set	Validation	Test set	Image size	object size
Banana	(133, 642)	(28, 128)	(28, 134)	3024×4032	461(1.24)
Cucumber	(21, 457)	(3, 75)	(4, 118)	6000×4000	279(1.07)
Avocado	(17, 613)	(2, 110)	(5, 143)	3024×4032	126(0.59)
Tomato	(22, 572)	(3, 107)	(6, 173)	5184×3456	227(0.94)
Tomato whole plant	(30, 223)	(7, 82)	(10, 198)	6000×4000	963(0.89)

work is wider in scope. Specifically it includes comparison of several (newer) networks, a comparison to human performance, detailed analysis of the main error causes: occlusion and scale, and consideration of performance measurement beyond the general F_1 or AP statistics.

3 Method

The datasets used in this work are briefly presented in 3.1, followed by a short description of the networks used in 3.2. In Section 3.3 image resolution and detection with multiple scales are described. Section 3.4 describes human performance estimation and 3.5 discusses agriculture-related performance measurements.

3.1 Datasets

Datasets were collected and annotated for five different crops: banana bunches, cucumbers, avocados, tomato fruits, and tomato plants. Images sizes differ between datsets in the range of 12-24 mega pixel. The number of objects per frame varies between 4 up to 72 objects. For detailed information on the number of objects and image sizes see table 1. As seen in figure 1, the images include large scale and occlusion variation, with some of the dataset (tomato, cucumber, avocado), dominated by the large amount of far and small objects. The challenge in shape and color varies between dataset: cucumbers are often small and hard to differentiate from branches. Tomatoes change color before harvesting, but the majority of the tomatoes in our data are unripe and therefore green and blending in with the green background of the foliage. Tomato whole plants have an irregular non-convex shape making it harder to demarcate one from the other.

3.2 Detection models

We tested three state-of-the-art detection algorithms: Mask R-CNN [9], RetinaNet [14], and EfficientDet [23]. While there are significant differences, all these networks share a common general structure. First, a pre-trained classification network, termed the 'backbone' network, is applied in a fully-convolutional manner to produce a dense representation for the entire image. At a second stage a variant of Feature Pyramid Network (FPN) [13] is applied. It creates tensors of similar representation but different resolutions, representing the image in multiple octaves to enable multiple scale detection. The model then tests for object existence in a pre-defined set of (position, scale) candidate rectangles termed 'anchors', which typically contains hundreds-of-thousands of candidates. The candidates, or a filtered subset of them, are then passed to processing by several parallel 'head' modules. A classification head is trained to classify candidates among non-objects and classes of interest. A second 'bounding-box refining' head is trained to refine the proposed rectangle, in case it contains an object, to a tighter rectangle with better fit to the object extent.

Mask R-CNN [9], has evolved as an improved variant of Faster R-CNN [18] with optional object segmentation capabilities. This is a two-stage model: the first stage, termed a Region Proposal Network (RPN) [18], filters from the possible anchors a few hundreds/thousands for further processing. It uses a ResNet-50 [10] backbone network to produce the initial representation, and the FPN, to create the multi-scale pyramid representation. An object/non-object initial classification is made for each anchor for the filtering. While positive object proposal are carefully chosen, negative 'no object' candidates (required for classification training at the second stage), are chosen heuristically to balance the number of positives. The object candidate regions are sampled from the representation tensors using a sampling layer (RoI-Align) and sent to the second stage, which includes the classification and bounding-box refining heads. There is no gradient flow between the stages in training, and they are essentially trained separately.

RetinaNet [14] is similar to Mask R-CNN in its usage of ResNet-50 network as backbone, and the FPN [13] for multiple scale representation. However, unlike Mask R-CNN, this is a single stage network trained end-to-end. Instead of filtering object candidates, all the hundreds-of-thousands anchors are considered as candidates and go through the object classification and bounding box regression heads. While enabling end-to-end training, this creates a problem of class imbalance, as classification is trained with hundreds of thousands of negative examples (non object candidates) versus a few positive examples in each image. The problem is addressed using a modification to the standard cross entropy loss termed 'Focal Loss', in which 'easy examples', including most of the negatives, are down-weighted in training. This mechanism channels the network learning effort efficiently to the hard examples, both positive and negative.

EfficientDet [23] is a one-stage network similar to the RetinaNet, and like it uses the focal-loss in training. However, it includes several improvements, and was recently (2019) reported to achieve state-of-the-art results on the MS-COCO detection challenge. The backbone used in this model is the B4-EfficientNet [22], reported to have higher ImageNet accuracy than ResNet-50 while using only one fifth of the parameters and running 10× faster. A second module in which significant changes were made is the FPN. EfficientDet uses a modified version termed Bi-FPN, which includes top-down connections between consecutive resolutions, and a weighting mechanism for fusion of information in these connections.

3.3 Image resolution

The datasets' images are typically very large (see table 1). The networks cannot accept such resolutions due to GPU memory limitations, and are limited to 1024×1024 input size. In a simple treatment, each image is resized such that its larger dimension is resized to 1024 pixels, keeping the aspect ratio, and padded with zeros in the shorter dimension. This down-scaling clearly diminish the object's size by a significant factor. For example in the cucumber dataset the scale factor is $max(\frac{6000}{1024}, \frac{4000}{1024}) = 5.85$, so each object's size is $5.85X$ smaller and their areas is $34X$ smaller than in the raw data.

We overcame this issue by working with images in two resolutions. Instead of using only the down-sampled original image, a set of 1024×1024 sub-images covering the original image were cropped with a fixed overlap. Both the down sampled original image and the cropped sub-images are used in network training and inference. As will be discussed below, both resolutions are required, and this two-resolution policy provides the detectors more opportunities to detect an object either in the sub-images or in the resized original image. The detected bounding boxes set of both resolutions are unified before Non Maxima Supression (NMS). The overlap parameter was chosen using initial empirical tests with the tomato datset and was set to 581 pixels. Note that with such overlap, which is close to half the sub-image size, each object is typically seen in 4 sub-images and one time in the single full-image, so the system has 5 opportunities to detect it.

An analysis of the effect of object scale on detection performance is presented in figure 2. The detection bounding boxes of the Mask R-CNN model for the tomato dataset were used to compute several statistics of interest. It turns out object scale has a profound impact on detection performance. Larger objects are more likely to be detected, have higher IoU (Intersection over Union) with the correspond ground truth rectangles, and higher confidence scores. Specifically detection probability arises linearly with object scale (measured logarithmically) in a significant scale range. A particular statistic of interest in our system is the 'second chance' probability: the probability to find an object in its larger scale (the sub-images) given that its detection has failed in the down-scaled image. Surprisingly, this probability is high not only for the small objects, but more for medium size objects, where it gets to values in $[0.4 - 0.5]$.

Since there are many sub-images and only few full downscaled images (the relation between them is 77:1 for the tomato datatset), the former dominate the dataset statistics. This sometimes create a problem in training, since large objects usually appear in the sub-images as partial objects. A very big object is typically seen in the data once as a full object in the full image, and 4 times as a partial object in sub-images. Upon training, This creates a tendency of the models to detect large objects with multiple bounding box corresponding to their parts, which is detrimental to performance. To avoid this tendency, we use two means. First, an object is annotated in a sub-image only if at least 60% of it is visible. Second, the full downscaled images are assigned a higher weight ($8\times$ higher) in training, to enhance the importance of detecting whole objects.

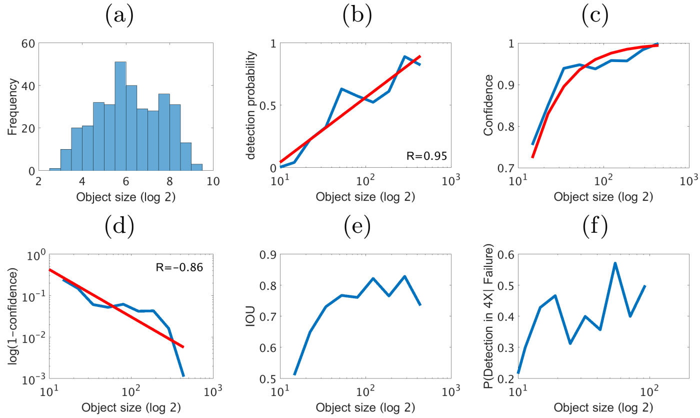


Fig. 2. The relation between object scale and detection quality. The graphs show empirical analysis conducted on detected bounding boxes of a tomato Mask R-CNN model. object scale is measured as $\sqrt{height \cdot weight}$ in logarithmic scale with base 2. (a) Detection scale histogram. (b) Detection probability as a function of object scale. A close-to-linear relation holds for a significant scale range. The linear approximation is shown in red. (c) network detections' confidence as a function of scale. The red line shows the model based on (d). (d) $log(1 - confidence)$ as a function of scale. the red line shows a linear model fitted. (e) IoU with object rectangle as a function of scale. It rises, then saturates for large objects. (f) The probability for an object to be detected in the sub-images given that it was not detected in the original down-scaled image

3.4 Analysis and human performance estimation

In a difficult detection task as presented here humans do not usually reach perfect performance. Specifically severe occlusion cases and small far objects can be easily missed, and foliage's texture creates false alarms. In addition, annotators are different in their skills and capabilities. As a fact, different annotators produce very different annotations when annotating the same dataset, as can be seen in Table 2. While we do not know which annotator is better, human performance can be measured by checking the degree of agreement between annotators. Specifically, we can define the task as predicting the annotations of a specific human, and compare a network to other humans in this task.

A comparison between algorithms and humans is hence made by temporarily setting one annotator as the Ground Truth (GT) annotator. The other human annotators are considered 'detectors', and are measured just like a detection algorithm would. However, human annotators do not provide confidence score for their annotations, so a recall-precision curve cannot be plotted for them and an AP score cannot be computed. Instead they provide a single (recall, precision) working point. While AP cannot be computed, an F_1 score for the (recall, precision) point can be computed and compared to the best F_1 score obtainable by a competing algorithm. The competing algorithms are trained on mixed annotation by sampling randomly a single annotator per image at training. All the non-GT annotators and the algorithms are evaluated and compared on the same test set. Since no human annotator is a-priory preferable to the others, the process is repeated for every human annotator at the GT annotator role.

Table 2. The number of annotations for each annotator and dataset. Tomato and cucumber datasets were annotated by three annotators, banana and avocado by two

Data-set	First annotator	Second annotator	Third annotator
Banana	118	139	—
Cucumber	119	108	127
Avocado	143	141	—
Tomato	212	215	149

A different analysis direction is to evaluate detector performance on object subsets of interest. Specifically, we suggest to compute the recall precision curve in six sub-categories of objects: all, small, big, occluded, non-occluded, big and non-occluded. Such category breakdown can helps a lot in understanding the model's strengths and weaknesses. Moreover, it enables understanding of the model capability to perform certain applications. For example, detector usage as the first stage for phenotyping only required success for fully visible and big objects where the phenotype can be measured. Such sub-category analysis required additional annotation, and it was performed for a single dataset - the tomato set. For size, a threshold between small and big objects was chosen at $100.5 Kpixels$ based on manual inspection, keeping a portion of 24.55% as big objects. Object occlusion was determined by manual annotation.

The recall precision curve for such a subset of interest cannot be measure with standard recall and precision definitions. The reason is that the classifier of interest was trained to detect all objects, not just the subset. If the set of ground truth object is trimmed to a subset, for example of small objects only, than detector hits on the complementary set (big) are considered false alarms, leading to low and irrelevant precision rates. To avoid this, one has to keep using the full object set in the false alarm definition. Formally, denote the full set of ground truth rectangles by GT, and the subset of interest by S. A detection rectangle D is now defined to be True Positive (TP) or False Negative (FN) by

$$\begin{aligned} D \in TP \ \ iff \ \ \exists R \in S \ \ s.t \ \ IoU(R,D) > 0.5 \\ D \in FP \ \ iff \ \ \sim \exists R \in GT \ \ s.t \ \ IoU(R,D) > 0.5 \end{aligned} \tag{1}$$

Hence the definition of TP has been narrowed to include only relevant objects, but the definition of false alarm keeps the original object set on which the classifier was trained.

The suggested definition enables measuring recall precision curves for subsets of interest, but it is not well suited when a single (True Positive Rate (TPR), False Positive Rate(FPR)) working point exist, as is the case with humans. Since the FPR is fixed across subsets, small subsets (hence with low TPR) obtain low precision $P = TPR/(TPR + FPR)$ and hence low F_1 scores. For comparison with human on subsets, we hence look at recall rates (of the algorithm and the human) at the same FPR, determined by the human working point.

3.5 Performance estimation for agriculture tasks

Detectors are commonly evaluated using the Average Precision (AP) score, which measured the area under the recall-precision curve and provides a robust and threshold-independent performance measure. However, for the task families common in agricultural applications this measure is too general, and more specific additional measurements can be more informative.

Robotic applications typically require high localization accuracy, to enable robotic interaction with the plant. Localization quality is not measured at all in the standard AP measure. For a single successful detection, localization accuracy can be measured by considering the center pixel deviation, i.e the distance between centers of detection and GT rectangles. This deviation is in pixel units, and can be divided by the GT object scale to get the relative deviation, which is a unit-less, more intuitive fraction. The relative deviation can be averaged over all successful detections and provide the mean relative deviation.

Another requirement in some robotic applications is finding all the objects, i.e. high recall, in order to perform the task for all of them (like harvesting or overcoming noxious entities). While high recall have the cost of low precision, most false alarms can be corrected by moving the robot closer to the object. Practically, we can measure the recall at 0.1 precision as an estimate of this 'total recall', measuring the fraction of relevant objects found by the detector.

In counting applications, the detector is typically applied with a certain threshold and its output rectangles are counted. To measure performance, count deviation from the true count is measured, and divided by the true count to get the relative count error - the deviation as a fraction of the true count. The natural threshold to use is the one without bias: the one for which the expected number of false positives (non objects identified as objects) is equal to the amount of false negatives (objects not identified). In that case, the counting error expectation is zero. Hence we propose to use the average relative count error, of the count estimates at the non biased threshold.

In Detection-based phenotyping, detectors are used as a first stage to enable phenotype measurements(e.g. [1, 24]). The detector finds the objects, then another model measures the desired feature. Typically the breeder is interested in statistics of the feature across a field or plot, like average and std of cucumber lengths [24] or spikelet count in a wheat spike. For estimation of such statistics, the detector does not have to consider all objects: a small sample of 'measurable objects' is enough. However, false positives are harmful, as each FP detection produces a 'noise' measurement contaminating the statistics. Therefore an appropriate detector measure will be the recall at 0.99 or 0.9 precision, where a minimal number of FP occur.

Table 3. Average Precision (AP) on the test set for each crop and network model

Data set	Mask R-CNN	RetinaNet	EfficientDet
Banana	0.741	0.604	0.455
Cucumber	0.507	0.516	0.453
Avocado	0.801	0.774	0.714
Tomato	0.580	0.646	0.522
Tomato whole plant	0.718	0.703	0.443

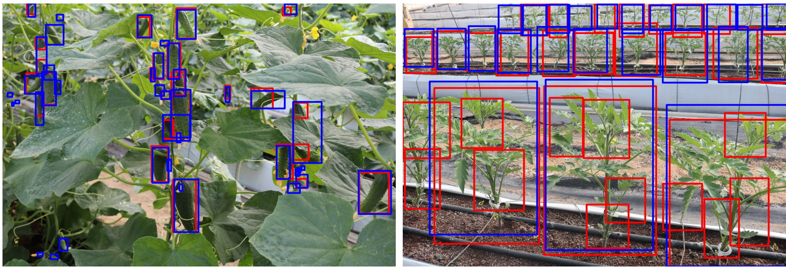
4 Experimental Results

We start by comparing the results of the three tested models on the five datasets. The best models are compared to human performance using F_1 scores. We continue with analysis of the obtained AP using the break-down of performance into object sub-categories. Finally, the additional agriculture-related performance measurements are reported and discussed.

Networks results: Table 3 shows the results of the three tested networks, trained with the dual-resolution approach. As can be seen, the comparison didn't yield a superior model, but Mask R-CNN and RetinaNet performed better than EfficientDet over all crops. Mask R-CNN works better on the 'easier' (as indicated by the obtained accuracy) datasets avocado, banana, and tomato whole plant, and RetinaNet performed better for the more difficult cucumber and tomato datasets. The results show similarity between the two leading models indicate that accuracy is primarily a function of dataset difficulty, not of chosen network. Surprisingly, scale variance, and not only mean scale, is a prominent source of difficulty. The avocado data, despite having the smallest mean object size was successfully handled, probably because is has low object size variance (see table 1) and a rather fixed view point across images. Banana and tomato whole plant, which are larger and do not suffer from high occlusion rates are of medium difficulty. The most difficult are tomato and cucumber, which are small, have high scale variance, and severe occlusion problems.

Comparison to human performance: Table 4 shows comparisons between the best trained models and a human detector, in terms of obtained F_1 scores. For the networks, the (recall, precision) working point with the highest F_1 score was chosen for comparison. The results show that for most tasks, a significant gap still exist between human and network detection. For Banana bunches, the network practically achieves human level detection: its agreement with human annotators is similar to the agreement between themselves. For Avocado, the network achieves high detection rates, yet humans are approximately 10% better. For the difficult tomato and cucumber data, significant gaps of $30-40\%$ exist. The reasons for this bug gaps and analyzed next.

Error cause analysis: Recall-Precision graphs for object subsets of the tomato dataset are shown in figure 4. Graphs are plotted for 6 subsets based on occlusion (occluded/non occluded) and scale (small/big) binary variables defined



(a) (b)

Fig. 3. Detection results examples: Red bounding-boxes are the detection results and blue bounding-boxes represents the ground truth annotations. Additional examples can be found in appendix 6

Table 4. F_1 scores obtained for (data set, annotator) tasks, with the annotator index defining the ground truth. The best model chosen based on the AP scores from table 3 is compared to the human annotators. The model type (Mask-R-CNN or RetinaNet) is indicated by (R) or (M) in parenthesis. Duplicate figures for human F_1 score are due to the symmetry of using one annotator to estimate the other and vice-versa

Crop, annotator	Model F_1	Human F_1	Human F_1
Banana, 1st	0.826 (M)	0.794	—
Banana, 2nd	0.748 (M)	0.794	—
Cucumber, 1st	0.589 (R)	0.789	0.802
Cucumber, 2nd	0.543 (R)	0.789	0.791
Cucumber, 3rd	0.548 (R)	0.802	0.791
Avocado, 1st	0.801 (M)	0.894	—
Avocado, 2nd	0.824 (M)	0.894	—
Tomato, 1st	0.64 (R)	0.903	0.923
Tomato, 2nd	0.648 (R)	0.903	0.905
Tomato, 3rd	0.636 (R)	0.923	0.905

in section 3.4. The results clearly reveal scale and occlusion as the dominant causes of detection errors, and quantify their impact. Specifically, Moving from big to small objects causes 25% degradation in accuracy (from 0.797 AP to 0.6), and introducing occlusion degrades accuracy by 30% (from 0.82 to 0.578). The detection accuracy is practically perfect for big non-occluded objects, where these two causes of difficulty are gone. The latter result encourages the usage of detectors to extract objects for secondary phenotyping measurements, where only measurable un-occluded objects are of interest.

In table 5 accuracy of networks and humans is compared according to scale and occlusion sub-categories. Occlusion and scale are the error causes of both human and the network, but the degradation form is different and more severe for the network. It can be seen that humans keep close to perfect performance as long as the object is either big or non-occluded, so their errors arise when objects

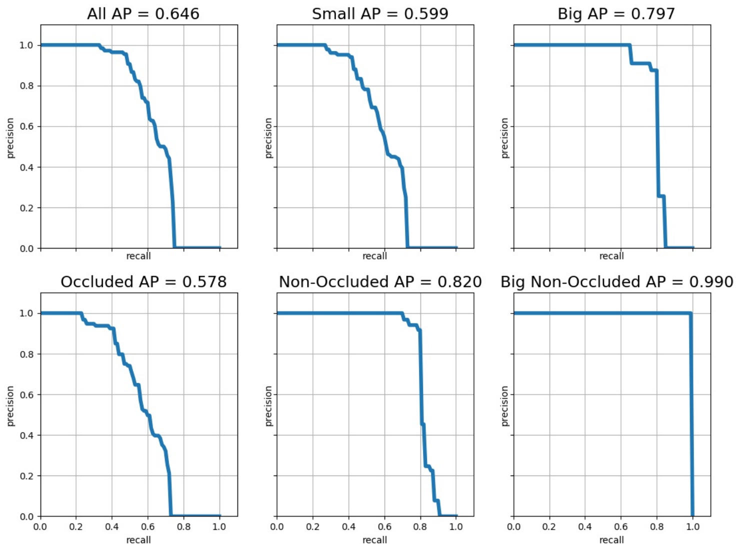


Fig. 4. Recall-Precision graphs of the RetinaNet tomato model on object subsets of interest. Average Precision (AP) scores are stated in the titles. Similar graphs for the other datasets, but only for big/small division, may be found in Appendix 2

are both small and occluded. The network significantly degrades and looses half of the recall rate due to size and occlusion independently.

Agriculture-related performance measurements: The performance indices from section 3.5 were measured for the best models. The results, presented in table 6, give rise to several observations. For counting, relative deviation depends not only on detector accuracy, but also on the typical number of objects per image. With more objects per image, relative count deviation gets smaller due to the law of large numbers. Hence better accuracy is obtained mainly for datasets with high number of objects per image (see table 1), like cucumber and avocado. For detection-based phenotyping, were a sample is required with minimal number of false alarms, the results indicate significant maturity of current detectors. With 1% of false positives, the detectors can sample $13 - 43\%$ of the objects, and if a noise of 10% false alarms can be tolerated most detectors can retrieve more than half of the objects. For localization the results are encouraging, with relative deviation lower than 14% obtained for 3 of the 5 datasets. However, since robotic applications require accuracy mainly for near (hence big) objects, characterization of localization error as a function of scale is required.

Table 5. Recall rates at specific False-Positive Rates (FPR) for network and humans on tomato dataset. Each row considers a different object subset. Columns present results obtained when different annotators are providing the ground truth. The average FPR of the Non-GT annotators is stated in the columns title in parentheses. Recall rates of the RetineNet model at the same FPR are reported in the 'Network' columns. Rates reported in 'Human' columns are the average recall of the two non-GT annotators

Category	Annotator 1 (16)		Annotator 2 (17)		Annotator 3 (15)	
	Network	Human	Network	Human	Network	Human
Small	0.46	0.90	0.48	0.89	0.43	0.90
Big	0.79	0.98	0.77	0.98	0.85	1
Occluded	0.44	0.89	0.47	0.90	0.41	0.90
Non occluded	0.79	0.98	0.79	0.98	0.79	1
Big non-occluded	1	1	1	1	0.89	1

Table 6. Performance measurements suggested in section 3.5, measured for each crop by the best model

Measurement	Banana	Cucumber	Avocado	Tomato	Tomato plant
Count deviation	0.452	0.117	0.152	0.328	0.354
Recall@0.99	0.430	0.22	0.13	0.34	0.11
Recall@0.9	0.680	0.31	0.68	0.51	0.52
Recall@0.1	0.780	0.62	0.89	0.75	0.85
Localization dev.	0.341	0.221	0.133	0.139	0.184

5 Concluding remarks

The benchmark reveals that current detection networks are able to achieve human level accuracy for banana bunch detection, and get close to this level for avocado. However in more difficult tasks significant gaps exist. The two dominant causes of error were identified to be small object scale and occlusion. Each of these variables causes performance degradation of $25-30\%$, and when these are removed detection is nearly perfect. The results were obtained with relatively small samples and may clearly improve with data size, but they nevertheless suggest clear directions for focusing work on detectors improvement. The task-related performance measurement show high potential for counting and detection-based phenotyping. For counting, 14% error were obtained for several datasets, even without usage of further mechanisms common in counting networks. For detection-based phenotyping current detectors were shown to be mature enough, as they are able to provide representative samples with high precision, and detect almost flawlessly big and un-occluded objects.

References

1. Baharav, T., Bariya, M., Zakhor, A.: In situ height and width estimation of sorghum plants from 2.5 d infrared images. Electronic Imaging **2017**(17), 122–

135 (2017)
2. Bargoti, S., Underwood, J.: Deep fruit detection in orchards. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). pp. 3626–3633. IEEE (2017)
3. Berenstein, R., Shahar, O.B., Shapiro, A., Edan, Y.: Grape clusters and foliage detection algorithms for autonomous selective vineyard sprayer. Intelligent Service Robotics **3**(4), 233–243 (2010)
4. Costa, C., Schurr, U., Loreto, F., Menesatti, P., Carpentier, S.: Plant phenotyping research trends, a science mapping approach. Frontiers in plant science **9**, 1933 (2019)
5. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International Journal of Computer Vision **88**(2), 303–338 (Jun 2010)
6. Farjon, G., Krikeb, O., Hillel, A.B., Alchanatis, V.: Detection and counting of flowers on apple trees for better chemical thinning decisions. Precision Agriculture pp. 1–19 (2019)
7. Fuentes, A., Yoon, S., Kim, S.C., Park, D.S.: A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. Sensors **17**(9), 2022 (2017)
8. Gongal, A., Amatya, S., Karkee, M., Zhang, Q., Lewis, K.: Sensors and systems for fruit detection and localization: A review. Computers and Electronics in Agriculture **116**, 8–19 (2015)
9. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
11. Kamilaris, A., Prenafeta-Boldú, F.X.: Deep learning in agriculture: A survey. Computers and electronics in agriculture **147**, 70–90 (2018)
12. Li, P., Lee, S.h., Hsu, H.Y.: Review on fruit harvesting method for potential use of automatic fruit harvesting systems. Procedia Engineering **23**, 351–366 (2011)
13. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
14. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)
15. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision – ECCV 2014. pp. 740–755. Springer International Publishing, Cham (2014)
16. Linker, R.: A procedure for estimating the number of green mature apples in night-time orchard images using light distribution and its application to yield estimation. Precision Agriculture **18**(1), 59–75 (2017)
17. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)
18. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. pp. 91–99 (2015)

19. Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T., McCool, C.: Deepfruits: A fruit detection system using deep neural networks. Sensors **16**(8), 1222 (2016)
20. Santos, T.T., de Souza, L.L., dos Santos, A.A., Avila, S.: Grape detection, segmentation, and tracking using deep neural networks and three-dimensional association. Computers and Electronics in Agriculture **170**, 105247 (2020)
21. Schertz, C., Brown, G.: Basic considerations in mechanizing citrus harvest. Transactions of the ASAE **11**(3), 343–0346 (1968)
22. Tan, M., Le, Q.V.: Efficientnet: Rethinking model scaling for convolutional neural networks. arXiv preprint arXiv:1905.11946 (2019)
23. Tan, M., Pang, R., Le, Q.V.: Efficientdet: Scalable and efficient object detection. arXiv preprint arXiv:1911.09070 (2019)
24. Vit, A., Shani, G., Bar-Hillel, A.: Length phenotyping with interest point detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 0–0 (2019)
25. Vitzrabin, E., Edan, Y.: Adaptive thresholding with fusion using a rgbd sensor for red sweet-pepper detection. Biosystems Engineering **146**, 45–56 (2016)
26. Xiong, H., Cao, Z., Lu, H., Madec, S., Liu, L., Shen, C.: Tasselnetv2: in-field counting of wheat spikes with context-augmented local regression networks. Plant Methods **15**(1), 150 (2019)
27. Zheng, Y.Y., Kong, J.L., Jin, X.B., Wang, X.Y., Su, T.L., Zuo, M.: Cropdeep: the crop vision dataset for deep-learning-based classification and detection in precision agriculture. Sensors **19**(5), 1058 (2019)

6 Appendix 1 - Additional Detection results

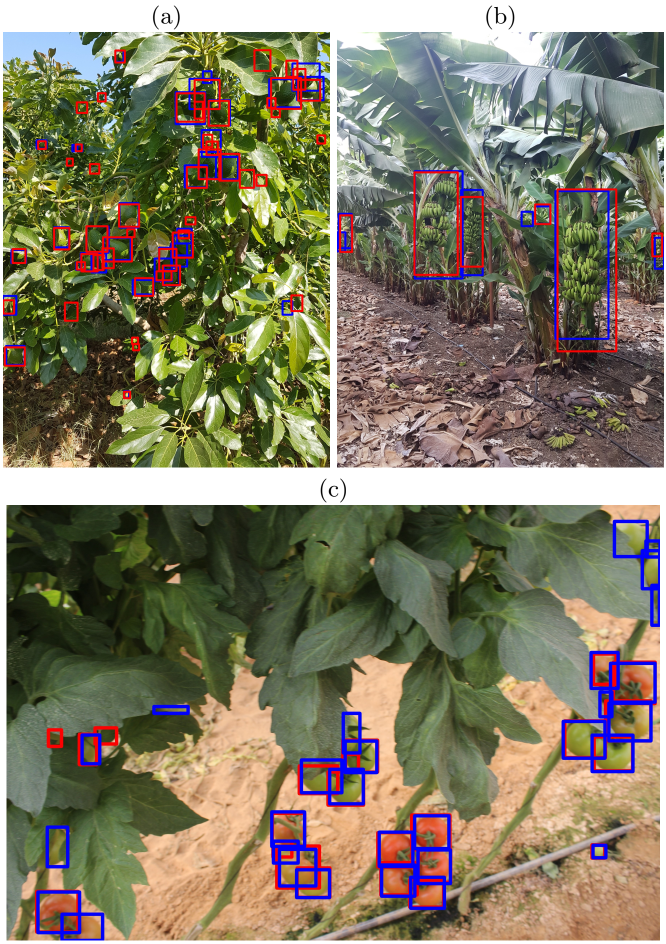


Fig. 5. Detection results examples: Red bounding-boxes are the detection results and blue bounding-boxes represents the ground truth annotations

7 Appendix 2 - Scale dependent Recall-Precision graphs

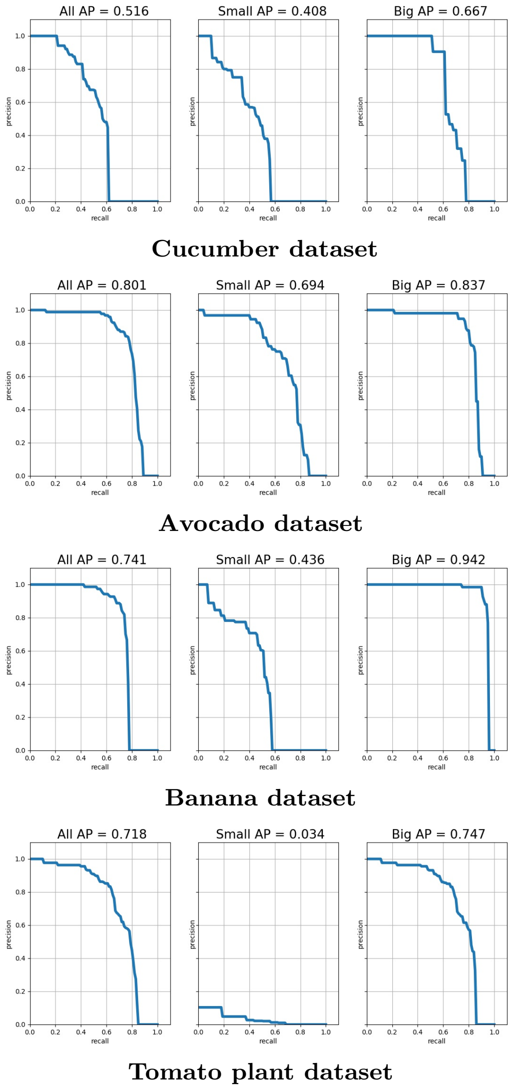


Fig. 6. Recall-Precision curves for subsets related to object scale difference. In each row the left curve is drawn for all the test set, the center curve for small objects, and the right one for big objects. Avocado scale threshold is different and smaller, since using the default 100, 500 pixels threshold would mark all objects as small. It was hence set to $20,000[pixels]$