

Image-Based Sorghum Head Counting When You Only Look Once

Lawrence Mosley
Omni Analytics Group
lawrence@omnianalytics.io

Hieu Pham
University of Alabama in Huntsville
hieu.pham@uah.edu

Eric Hare
Omni Analytics Group
eric@omnianalytics.io

Yogesh Bansal
Omni Analytics Group
yogesh.bansal@live.com

Charity Mwanza
University of Alabama in Huntsville
cm0260@uah.edu

Abstract

Modern trends in digital agriculture have seen a shift towards artificial intelligence for crop quality assessment and yield estimation. In this work, we document how a parameter-tuned single-shot object detection algorithm can be used to identify and count sorghum heads from aerial drone images. Our approach involves a novel exploratory analysis that identified key structural elements of the sorghum images and motivated the selection of parameter-tuned anchor boxes that contributed significantly to performance. These insights led to the development of a deep learning model that outperformed the baseline model and achieved an out-of-sample mean average precision of 0.95.

Keywords: Deep learning, computer vision, object detection, agriculture

1. Introduction

Phenotyping and genotyping comprise two main areas of plant breeding. Phenotyping involves the measurement of an observable trait, whereas genotyping studies the genetic composition of plants. While recent technological advancements have made genotyping more accurate, faster, and affordable, phenotyping has become the bottleneck in accelerating breeding programs (Tester and Langridge, 2010). Certain phenotypic traits, such as yield, are accurately and efficiently measured during the harvest process. However, other phenotypes, such as stalk strength and head count, require labor-intensive, expensive, and error prone manual intervention. The advent of modern technology combined with the need to innovate has created a cultural shift towards digital agriculture. The

use of self-driving equipment, as well as drone imagery and object recognition software, provide a glimpse towards the future of agriculture (Cariou et al., 2010; Thomas et al., 2021; Tripicchio et al., 2015).

Image-based algorithms for the detection and counting of crops have been applied to corn, grapes, tomatoes, apples, and mangoes, but these approaches typically require high-resolution images captured with minimum signal to noise ratios (Gnädinger and Schmidhalter, 2017; Khaki et al., 2022; Qureshi et al., 2017). With advancements in unmanned aerial vehicles, drone imagery, and machine learning, we are able to progress towards a digital future where labor-intensive phenotyping is no longer required (Shaikh et al., 2022). For the future of agriculture, merging these components will allow for a low cost way to monitor or estimate crop yield to identify low performing areas and detect damaged crops. For a farmer who manages more than 10,000 acres of land, traversing each field is not a feasible option. The ability for captured aerial field images to alert farmers of any in-field phenotypic variation is vital for making real-time decisions on managing fields.

In the case of sorghum (*Sorghum bicolor*), being able to monitor the growth stages and color of their heads informs the farmer of the health and quality of their crop. With this information, farmers are able to manage decisions on how to maximize the growth potential of their sorghum by possibly spraying pesticides, adding fertilizers, etc. Current practice leans on manual, labor intensive counting of heads in a field. For small to medium sized operations, this may not present a significant burden. However, in commercial breeding programs and traditional farms, the number of sorghum plots often exceeds thousands of acres. The sheer number of plots makes it infeasible to accurately

count all heads, limiting the effectiveness of manual approaches. This manual labor bottleneck is one of many motivating factors for combining drone imagery and machine learning. Taken by an autonomous drone, Figure 1 is an example overhead shot of two rows of sorghum crops. Instead of traversing hundreds of acres of land to gather data and crop characteristics manually, commercial breeding programs and traditional farmers can utilize advancements in technology to enable real-time decision making.



Figure 1. Image of sorghum grass taken with a UAV. The individual heads are the circular mild yellow-ish objects resting on top of the green curvi-linear leaves.

Specifically, for this paper,

- we aim to provide an approach to count sorghum heads by way of the single-shot “You Only Look Once” object detection algorithm (Bochkovskiy et al., 2020), and
- we provide insights into the data through that help motivate the tuning of YOLO parameters through detailed exploratory data analysis.

For simplicity, we will be referencing the “You Only Look Once” Version 4, YOLOv4, implementation for this work, and for simplicity, will simply refer to this as YOLO for the duration of this paper. We demonstrate that accurate detection results for plant phenotyping, in particular sorghum head detection, are obtainable through anchor box tuning and can thus mitigate the bottleneck imposed by manual counting methods. Additionally, we provide insights into the data through that help motivate the tuning of YOLO parameters. To achieve this goal, section 2 provides an exploratory analysis on the sorghum image. Section 3 outlines the methodology of our approach, while results. Lastly, the discussion is elaborated in Section 4 before concluding with Section 5.

2. Data and Exploratory Methods

350 aerial images of sorghum heads were used for this analysis, separated into 300 labeled training and 50 labeled test images. Structurally, each training instance is a duple containing both an image and structured text file containing the locations, in pixels coordinates, of the

individual bounding boxes for each identified sorghum head in the image. This structure, found in Table 1, defines the class and the four edge coordinate locations of the box containing the plant head. These coordinates can then be linked back to the source image, such as the one seen in Figure 1

Table 1. Sample bounding box coordinates for sorghum heads in the training set.

Class Name	Left	Top	Right	Bottom
sorghumHead	16	618	41	639
sorghumHead	33	1036	63	1067
sorghumHead	34	383	75	424
sorghumHead	42	1019	85	1059
sorghumHead	43	722	69	745
sorghumHead	44	952	73	976

To begin the analysis process, we seek to gain an understanding of the uniformity of the imagery data first by visual inspection. At both the image and individual sorghum head levels, we constructed image collages to facilitate within and across image analysis. For object detection tasks, identification accuracy is made often higher when the images conform to the same uniform lighting, perspective, and dimensional standards across all photos due to increased signal to noise ratios. However, for the images in our training sample, the collage in Figure 2 highlights variation within each of the aforementioned categories. That is, various studies have shown that the quality of the image as well as the consistency of the training images can greatly impact model performance (Cuhadar et al., 2021; Lin et al., 2019). In our use case, because this data is acquired live in-field, the optimal training data to have would be a large spread of images covering every possible environmental and lighting conditions that is representative of real in-field inference applications. However, in this application, and many other computer vision tasks, this is not possible and thus a limitation to work in this area.

Inspection shows that within the collection, there are differences in lighting and heights, unexpected fissures within rows, and obstructions blocking the view of the sorghum heads. These effects are more prevalent when scrutinizing the individual heads within an image. In the context of object detection, the non-uniform lighting conditions can be mitigated through a covering of training images. That is, as long as there are a sufficient number of sorghum heads which demonstrate the different lighting conditions, deep learning algorithms will still be able to perform well.

When focusing on the sorghum heads themselves,

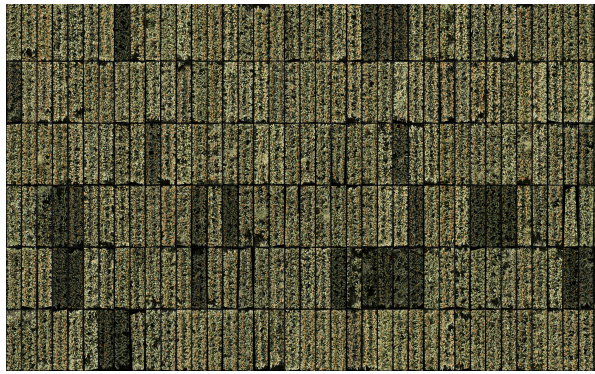


Figure 2. Collage of sorghum images highlighting the variation in photo quality.

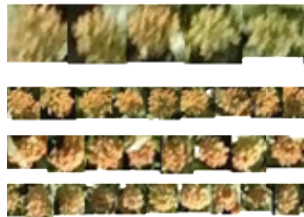


Figure 3. Collage of individual sorghum heads cropped from a random sample of training images.

we identified sources of variation in the sorghum heads that can be attributed to factors both endogenous to the plant and exogenous sources from the imaging procedure. At the sorghum head level, variation in the size of the individual sorghum heads, the plant height and the image quality of the individual heads, all point to a lack of uniformity across the images. Outside factors such as drone flight height and variation in landscape height can also artificially increase or decrease the size of the heads within an image, suggesting that some form of control or normalization within our modeling process will be necessary, as we will explain in Section 3.1. The sorghum head collage in Figure 3 showcases examples of image distortions, where identifying singular plant heads is complicated by the existence of low quality image sections which introduce noise inside of the pixel features our future classifier will extract.

To further quantify the quality of the input images, we analyzed the distributions of the number of sorghum heads in an image and the area of the image covered by sorghum. For the collection of 300 training images, there were 30,953 sorghum heads individually labeled with bounding boxes. Collectively, the distribution across all the training samples well approximates a Normal distribution with, 103 sorghum heads contained within each image, on average. This is illustrated in Figure 4.

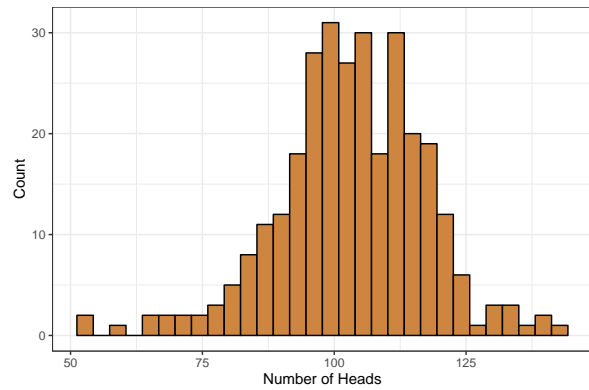


Figure 4. Distribution of the number of sorghum heads within the 300 training images.

The next step in our exploratory analysis is to understand the amount of sorghum present in each image. By taking the total area of the labeled bounding boxes within an image and normalizing by the dimensions, we were able to create an estimate of sorghum head coverage. This estimate was then used to identify outlying images that may have potential data errors. Shown in Figure 5, this metric highlighted a large variation in the sorghum coverage, but inspection of the individual images did not support the removal of any training instances. These metrics were constructed to deliver guiding intuition for the diagnoses of our modeling process by providing reasonable bounds for model outputs that could inform us whether the results for a training session were acceptable. Therefore, we conclude because of the seemingly normal distribution of the sorghum heads throughout the images that each of the training images should be retained. If it were the case that some images contained large outliers such as the entire image with 100% coverage of sorghum heads, we would have removed those images from our sample to not adversely bias our results. Additionally, creating these two visualizations provides us with an immediate check on the well-being of our trained model. For example, if we created a model that only identified three sorghum heads or a sorghum coverage of less than 5%, we would immediately identify an error in the model fitting process that would need to be remedied.

3. Object Detection

Current state of the art methods for object detection and counting leverage deep learning models to perform classification of an image or video (Falk et al. (2019) and Heinrich et al. (2019)). The specific architectures of these deep learning implementations vary, but all rely on multi-layered neural networks trained with GPUs

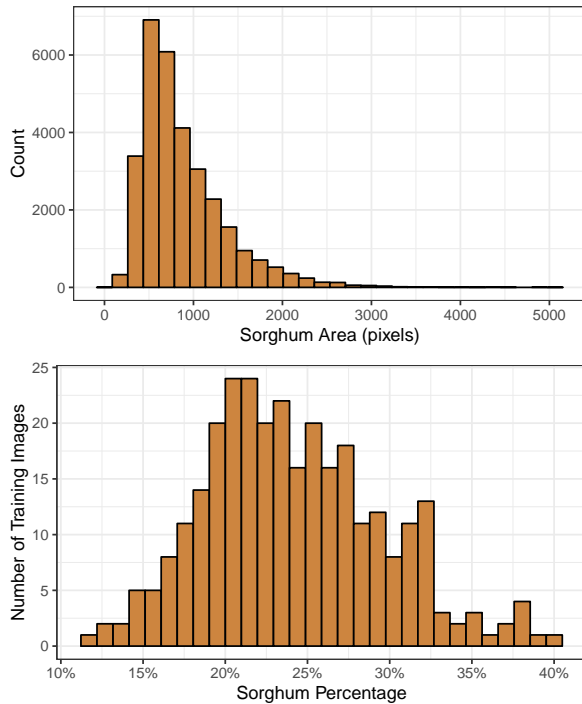


Figure 5. Distribution of sorghum areas and percentages, respectively.

on exceptionally large image datasets. To perform the sorghum identification and counting task outlined in this paper, the authors leveraged a task tuned version of the YOLO algorithm. Specifically for this sorghum counting task, we leveraged the pretrained weights from the YOLO’s Darknet-53 architecture, a model framework consisting of 53 convolutional layers trained on the ImageNet dataset for the purpose of feature extraction and classification.

YOLO is a full end-to-end convolutional neural network architecture, constructed to perform object detection through the recasting of the image classification problem as a bounding box regression task. Its approach takes a single image, partitions it into a user specified $S \times S$ square grid, and then attempts to identify objects by encapsulating each grid cell with the most appropriate bounding box and confidence scores for that box. This process is done simultaneously as a feature map of relevant visual constructs is learned by a neural network with a sequential arrangement of convolutional layers, residual blocks, and sampling layers. As a single-shot object detection algorithm, the YOLO model is configured to output an approximate location and the best fit bounding box from labeled images. During the training process, these images are first partitioned into grids where the deep learning

model can then focus on maximizing its ability to select and place anchor boxes, of the appropriate size, over all identifiable classes within each segment simultaneously. It is this process that not only gives single shot detection algorithms their state-of-the-art speed performance, but also makes their accuracy highly sensitive to the selection of the initializing anchor boxes.

To perform sorghum identification, using a domain neutral pre-trained YOLO model required modifications to the software implementation. By default, YOLO utilizes a multi-label classifier to identify objects; however, for this task, the configuration file was modified to account for the single sorghum class. As a result of there being only one class to identify, the intersection over the union for detection was set to 0.70 in order to force the model to return all potential sorghum heads, regardless of the associated class probability. Table 2 shows the configuration of the Darknet configuration file used for our final results.

Table 2. Parameters of YOLO Configuration.

Parameter	Value
Mask	1, 2, 3, 4, 5
Anchors	(10,10), (16,16), (19,19), (16,24), (24,20), (23,24), (28,27), (23,35), (32,32), (38,39), (50,50), (60,60), (80,80)
Classes	1
Number	13
Jitter	0.3
Ignore Threshold	0.7
Truth Threshold	1.0
Random	1.0

3.1. Tuning Anchor Boxes

During the weight updating process, at three separate phases, a collection of three anchor box offsets are predicted within each grid. An unconfigured, default YOLO configuration uses a k -means clustering method to search for nine candidate anchor boxes from the collection of bounding boxes found in the training set. The centroids of each group are then converted into rectangular coordinates to form the anchor boxes for which the final objects identified will be bound within. Motivated by the exploratory analysis done previously, we constructed a scatterplot of the height and width of all labeled bounding boxes to visually assess the quality of the algorithmically suggested anchor boxes. Figure 6. showcases our findings. The default solution did not propose rational anchor boxes that would sufficiently account for the variation in sorghum head size. The

anchor boxes selected via the k -means algorithm did not cover segments on the extreme lower tail of the bounding box distribution and, similarly, suggested multiple anchor boxes whose sizes exceeded the largest bounding areas found in the training set by a factor of three.

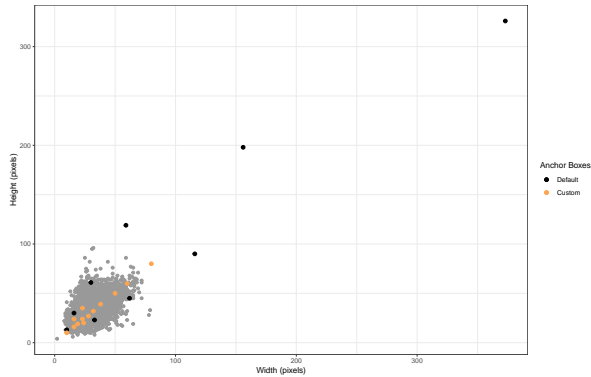


Figure 6. Bounding box dimensions, default, and tuned anchor box sizes in gray, black, and brown, respectively.

To overcome this deficiency, we approximated a linear model through the bounding boxes found in our training set and sampled nine anchor boxes along the line. For better coverage of smaller heads at the extreme lower tail of the bounding box distribution, we created a 10×10 anchor box. The remaining anchor boxes were placed in regions of above average variance. Highlighted in brown, the 13 anchor boxes used for prediction can be seen in Figure 7. As this figure demonstrates, we provide significantly better coverage than the default k -means algorithm anchor boxes. It is this trick that allows us to be able to detect sorghum heads of various sizes.

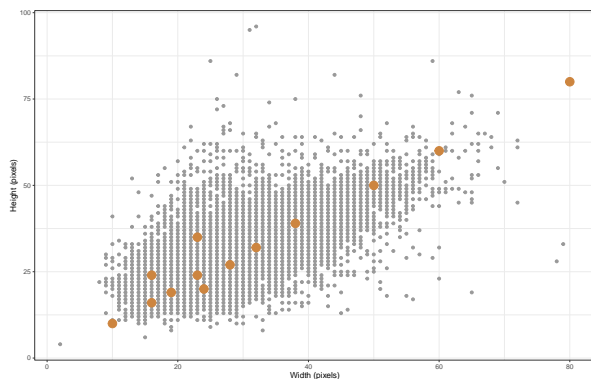


Figure 7. Zoomed-in bounding box dimensions and chosen anchor box sizes in gray and brown, respectively.

The final configuration adjustment to the original architecture consisted of assigning the anchor boxes to various up sampling layers where we included three boxes in the first layer, four in the second layer, and six boxes in the final layer. Lastly, to update the model weights, about 24 hours of additional training was performed on an Ubuntu 18.04.2 LTS workstation with 128 GB of memory, an Intel i7-6850k and a single NVidia GeForce 1080.

4. Results

In this section, we provide the summary of our YOLO implementation with parameter-tuned anchor boxes. However, before we begin our discussion on the results, we must first explain how we are measuring the task of detecting sorghum heads. For classification tasks, domain specific measures, which are often dependent on the structure of the data, are required to properly codify the concept of model performance. In object detection problems, evaluation is non-trivial because there are two distinct tasks:

1. Determining whether an object exists in the image
2. Determining the location of the object

We briefly discuss these two measures, but direct the reader to the cited references for more details. For measuring the existence of objects, a popular metric in measuring the accuracy of object detectors is the average precision (McFee and Lanckriet, 2010). Succinctly, average precision is the area under the precision-recall curve which visualizes the true positive rate and the positive predictive value for a predictive model (Davis and Goadrich, 2006). We then define mean average precision (mAP) as the average precision over all recall values. To measure the location accuracy of our predicted bounding box, we utilize the intersection over union (IoU) which summarizes how well the ground truth object overlaps the object boundary predicted by the model (Rezatofighi et al., 2019). Model object detections are determined to be true or false depending upon the IoU threshold, that is, an object is only detected if the IoU is above a certain level. If $\text{IoU} = 1$, then the ground truth bounding box is equal to the predicting bounding box in both size and locality. Whereas an $\text{IoU} = 0$ would indicate that the intersection of both boxes is empty. Since IoU provides a threshold for object detection, a lower IoU is likely to result in more objects being detected but lower precise localization.

The output of our object detection procedure resulted in separate files for each image with bounding box dimensions and a confidence score for every identified

object which is an estimated probability that the object is found within the predicted bounding area. This is shown in Table 3. This structure is akin to the one supplied with the training examples.

Table 3. Sample of identification confidence for sorghum heads in the test set. Note, confidence is abbreviated to “Conf.”.

Class Name	Conf.	Left	Top	Right	Bottom
sorghum	0.982	26	448	58	477
sorghum	0.978	73	790	104	819
sorghum	0.976	66	132	105	167
sorghum	0.975	231	538	266	568
sorghum	0.968	196	658	225	684
sorghum	0.966	242	627	275	656

For the labeled training samples, bounding boxes can be overlaid on the image to show deviations between the true and predicted locations of sorghum heads. This is illustrated in Figure 8 in red and blue, respectively. In the example images, the concepts of detection and intersection over union can be clearly observed. False negatives exist over sorghum heads that have true red labels, but lack a corresponding predicted blue bounding box label. For the sorghum heads found by the prediction procedure, the overlapping intersection between the coordinates give an assessment of the coverage. The incorrect localization of the predicted bounding box and the missed identification of sorghum heads were the dual inhibitors for achieving high mAP scores for our early modeling iterations. Moreover, we notice that the parameter-tuned YOLO model is better able to detect small sorghum heads. This is something that would not be attainable with the default anchor boxes. It is this flexibility in tuning the anchor boxes that allows for superior object localization than otherwise would be possible.



Figure 8. True bounding boxes and predicted bounding boxes; red and blue, respectively of default YOLO (top) and parameter-tuned YOLO (bottom).

Optimizing over both objectives, by utilizing the anchoring box adjustment technique discussed earlier, resulted in our final training score of 0.99 mAP and the score of 0.95 mAP on the final test set. Since our YOLO model is able to obtain a mAP of 0.99 in the training set, we take that to imply that the parameter-tuned anchor boxes do provide sufficient coverage for the sorghum heads as opposed to the default YOLO model. From Figure 9, it is immediately noticeable that there is indeed a noticeable improvement by tuning anchor boxes.

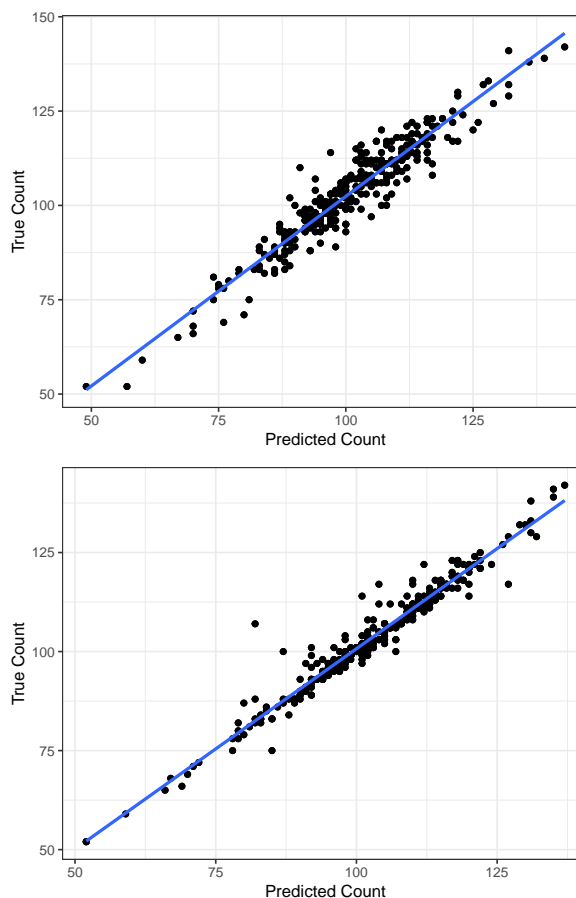


Figure 9. True count versus predicted count of default model and parameter-tuned anchor box model, respectively. The default model has an R2 value of 0.9016 while the parameter-tuned anchor box model has an R2 of 0.9513 (IoU threshold = 0.70 for both).

5. Discussion

We believe the main performance driver differentiating our parameter-tuned YOLO model from the default lies in the detection of smaller sorghum heads, which would otherwise be excluded by YOLO’s

embedded anchor box detection algorithm. The R2 value of the parameter-tuned YOLO model displays greater object detection precision than the default YOLO model given an IoU threshold of 0.70. Indeed, by performing our exploratory analysis, we were able to determine improved anchor box sizes as opposed to the default anchors chosen by k -means clustering, again by observing Figure 9.

Indeed, although the parameter-tuned anchor box performed superior to the default YOLO model on average, there were outliers. The most evident is the point where we predicted 78 sorghum heads but the true count was 108. For the default model, there is not such a discrepancy. This occurred simply because the optimized anchor boxes were too large for the specific image. That is, the anchor boxes were systematically determined by observing the size of all sorghum heads, not at the individual level. The simplest way to solve this issue is to increase the number of anchor boxes being considered from 13 to a larger number. This will ensure appropriate coverage. However, doing so will lead to increased computational complexity which may be an issue depending on the deployment use case (e.g. live in-field or post-processing). Because most of the sorghum heads across all images in our dataset are relatively the same size and contain the same camera conditions, we expect our model to perform well in all cases where similar image capturing settings are present.

The focus of this work is on YOLO because we wanted to showcase the power of tuning anchor boxes and how performance can increase by this one method. However, other object detection frameworks exist and can be used. One method is a RetinaNet, another object detection model similar to YOLO, used by (Gonzalo-Martín et al., 2021). Similar to our work, they augmented their data by observing geometrical and color features to create new observations to create an ensemble prediction model. From their original baseline model, their new approach increased accuracy by a comparable percentage to our parameter-tuned anchor box approach. However, their approach involves more parameter tuning and the complex creation of multiple models and weights. We consider this another strength of our straightforward approach in that we obtain superior results while maintaining an efficient parameter optimization strategy.

6. Conclusion and Future Work

In this work, we provide a novel example of the use of visual inference to inform the selection of tuning parameters for the improved accuracy of a one shot object detection model. Specifically, we show how

exploratory data analysis on the training bounding box sizes can provide a powerful context for assessing the quality of the anchor boxes selected for single shot object detection algorithms. Once tuned, the YOLO model architecture can be used for the identification and counting of phenotypic traits from aerial images. We are able to obtain an mAP of 0.95.

Previous trials suggest additional performance gains can be found through additional training epochs and further refinement of the anchor box selection. Extensions of this research will involve the pursuit of these alternative approaches. It is also possible to realize additional gains in accuracy by enlarging the solution space search by increasing the anchor boxes used and then training with high class GPUs or TPUs.

To expand to real-time detection of sorghum images and to support other high throughput phenotyping systems, a TinyYOLO implementation may be a computationally efficient enough model to be stored and run on a Raspberry Pi. This would allow for the automatic detection and counting of sorghum heads directly on the drones in real-time (Padala and Malathi, 2022; Pedoeem and Huang, 2018). Moreover, a future research direction could be to utilize the color and size of sorghum heads as an in-season measure of the quality of the plot. This would enable farmers to make real-time decisions without the need to manually inspect their entire field.

References

- Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.
- Cariou, C., Lenain, R., Thuilot, B., & Martinet, P. (2010). Autonomous maneuver of a farm vehicle with a trailed implement: Motion planner and lateral-longitudinal controllers. *2010 IEEE International Conference on Robotics and Automation*, 3819–3824. <https://doi.org/10.1109/ROBOT.2010.5509447>
- Cuhadar, C., Lau, G. P. S., & Tsao, H. N. (2021). A computer vision sensor for efficient object detection under varying lighting conditions. *Advanced Intelligent Systems*, 3(9), 2100055.
- Davis, J., & Goadrich, M. (2006). The relationship between precision-recall and roc curves. *Proceedings of the 23rd International Conference on Machine Learning*, 233–240. <https://doi.org/10.1145/1143844.1143874>
- Falk, T., Mai, D., Bensch, R., Çiçek, Ö., Abdulkadir, A., Marrakchi, Y., Böhm, A., Deubner, J., Jäckel, Z., Seiwald, K., Dovzhenko, A., Tietz, O.,

- Bosco, C. D., Walsh, S., Saltukoglu, D., Tay, T. L., Prinz, M., Palme, K., Simons, M., ... Ronneberger, O. (2019). U-net – deep learning for cell counting, detection, and morphometry. *Nature Methods*, *16*, 67–70. <http://lmb.informatik.uni-freiburg.de/Publications/2019/FMBCAMBBR19>
- Gnädinger, F., & Schmidhalter, U. (2017). Digital counts of maize plants by Unmanned Aerial Vehicles (UAVs). *Remote Sensing*, *9*(6), 544. <https://doi.org/10.3390/rs9060544>
- Gonzalo-Martin, C., Garcia-Pedrero, A., & Lillo-Saavedra, M. (2021). Improving deep learning sorghum head detection through test time augmentation. *Computers and Electronics in Agriculture*, *186*, 106179.
- Heinrich, K., Roth, A., & Zschech, P. (2019). Everything counts: A taxonomy of deep learning approaches for object counting.
- Khaki, S., Safaei, N., Pham, H., & Wang, L. (2022). Wheatnet: A lightweight convolutional neural network for high-throughput image-based wheat head detection and counting. *Neurocomputing*, *489*, 78–89.
- Lin, C.-w., Yeh, F.-m., Wu, B.-w., & Yang, C.-h. (2019). The effects of reflected glare and visual field lighting on computer vision syndrome. *Clinical and Experimental Optometry*, *102*(5), 513–520.
- McFee, B., & Lanckriet, G. (2010). Metric learning to rank. *Proceedings of the 27th International Conference on International Conference on Machine Learning*, 775–782.
- Padala, A. K. R., & Malathi, P. (2022). An optimized object detection system using you only look once algorithm and compare with tiny-yolo algorithm with increased accuracy. *2022 2nd International Conference on Innovative Practices in Technology and Management (ICIPTM)*, *2*, 606–610.
- Pedoeem, J., & Huang, R. (2018). YOLO-LITE: A real-time object detection algorithm optimized for non-gpu computers. *CoRR*, *abs/1811.05588*. <http://arxiv.org/abs/1811.05588>
- Qureshi, W. S., Payne, A., Walsh, K. B., Linker, R., Cohen, O., & Dailey, M. N. (2017). Machine vision for counting fruit on mango tree canopies. *Precision Agriculture*, *18*(2), 224–244. <https://doi.org/10.1007/s11119-016-9458-5>
- Rezatofghi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., & Savarese, S. (2019). Generalized intersection over union: A metric and a loss for bounding box regression. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Shaikh, T. A., Rasool, T., & Lone, F. R. (2022). Towards leveraging the role of machine learning and artificial intelligence in precision agriculture and smart farming. *Computers and Electronics in Agriculture*, *198*, 107119.
- Tester, M., & Langridge, P. (2010). Breeding technologies to increase crop production in a changing world. *Science*, *327*(5967), 818–822. <https://doi.org/10.1126/science.1183700>
- Thomas, G., Balocco, S., Mann, D., Simundsson, A., & Khorasani, N. (2021). Intelligent agricultural machinery using deep learning. *IEEE Instrumentation & Measurement Magazine*, *24*(2), 93–100.
- Tripicchio, P., Satler, M., Dabisias, G., Ruffaldi, E., & Avizzano, C. A. (2015). Towards smart farming and sustainable agriculture with drones. *2015 International Conference on Intelligent Environments*, 140–143. <https://doi.org/10.1109/IE.2015.29>