Fast ready-to-harvest cotton detection and classification with YOLOv8 in greenhouse crops

Guillem González[1], Antonio Martínez[1], Vicente Martínez[2], Sergi Foix[1], Guillem Alenyà[1]

Abstract—
Autonomous robotic systems come as a new way of harvesting cotton, which is needed to preserve quality while reducing expenses, as opposed to traditional methods. New algorithmic solutions must be developed to detect cotton and discern between ready-to-harvest and not ready-to-harvest bolls. Using YOLOv8 as our object detection model, this paper presents a new cotton boll image dataset, RTH-CONDIS (Ready-To-Harvest Cotton Discerning Imageset), with a total of 409 raw images, enlarged with data augmentation for the training process. After testing different sizes of YOLOv8, YOLOv8s is the most promising version for this project, with a final detection performance of 0.902 for mAP50, a recall of 0.852, and a precision of 0.901. As a result, we get satisfactory prediction metrics, considering the dataset's size. This solution is suitable for real-time, resource-limited implementations, as is needed for tracking and counting applications on a mobile harvester robot.

*Index Terms—***Cotton detection, cotton dataset, cotton ripeness classification, YOLOv8, greenhouse farming.**

I. Glossary

Please refer to Table I for acronym meanings.

TABLE I: Acronym glossary

Acronym	Full name
HVI	High Volume Instrument
DNN	Deep Neural Network
YOLO	You Only Look Once. n (nano), s (small) and m (medium)
SSD	Single Shot MultiBox Detector
mAP	Mean Average Precision
TAD	Type Augmented Dataset, referring to the original dataset augmented with some technique
OG	Original dataset
B	Brightness (factor range for change between -0.2 and 0.2)
R	Rotation (between -60º and 60º)
H	Flip-H (Horizontally)
BL	Blurring (random Gaussian kernel size applied between 5 and 35 pixels)
MN	Multiplicative Noise (random MN applied pixel-wise between 0.5 and 3.5 units) over RGB channels
COMB0	Combination of blurring and multiplicative noise techniques
COMB1	Combination of all techniques with stated parameters

II. Introduction

The global cotton use is estimated at almost 25,452 million metric tonnes for 2024/2025, the highest in the last 4 years, therefore being one of the most produced crops worldwide [1]. The current scenario for harvesting relies mainly on mechanical harvesting among developed nations. At the same time, in developing countries like India, multi-stage handpicking (by human labour) of cotton crops is widely used [2]. Machine-based methods offer fast, easy, and cheap harvesting compared to human labour, but they reduce cotton quality. New ways of collecting the cotton are needed.

On the other hand, in developed countries, cotton harvesting traditionally involved removing the entire plant, with cotton fibres later sorted from the rest. While this method is fast and cheap, the fibres can be damaged, resulting in lower quality compared to hand-picked cotton, which is more expensive [3].

The future of cotton cultivation in protected systems like greenhouses holds significant promise for sustainability by reducing water consumption, pesticide use, fertilizers and energy. Implementing new technologies is crucial for optimizing these production systems. Automating and robotizing the harvesting of cotton bolls, one by one will enhance fibre quality by preventing damage during picking, positively affecting HVI parameters such as fibre length, fineness, and strength. Additionally, by not destroying the plant, cotton can be grown perennially, maximizing annual production. Recent studies [4], [5] highlight the importance of these technological advancements for sustainable cotton farming.

Zhang et al. [6] demonstrated that plants continue producing cotton after the initial fibre harvest. Additionally, perennially cultivated crops, aged 2 to 4 years, perform equally well or better than annual cotton crops. Perennial cropping significantly reduces soil, nutrient, and water loss, while also lowering fertilizer demand [7]. Consequently, the cotton industry is interested in both avoiding replanting costs, due to continuous production after the first harvest and maximizing cotton yields.

Robotic cotton harvesting is a promising technology with the potential to improve harvesting efficiency, preserve cotton fibre quality, reduce yield losses, and promote sustainable cotton production [8].

This article describes the computer vision framework used for detecting, tracking, and counting cotton bolls in both ready-to-harvest and not-ready-to-harvest stages. Our cotton plants are located in a greenhouse (Fig. 1), but this study can be extrapolated to other forms of cotton farming. The objectives of this study are to:

1) Create a custom dataset for cotton boll ripeness classification.

[1] Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Llorens i Artigas 4-6, 08028 Barcelona, Spain; {aolivares,sfoix,galenya}@iri.upc.edu

[2] Department of Vegetal Nutrition, Centro de Edafología y Biología Aplicada del Segura del Consejo Superior de Investigaciones Científicas (CEBAS-CSIC), 30100 Murcia, Spain; vicente@cebas.csic.es

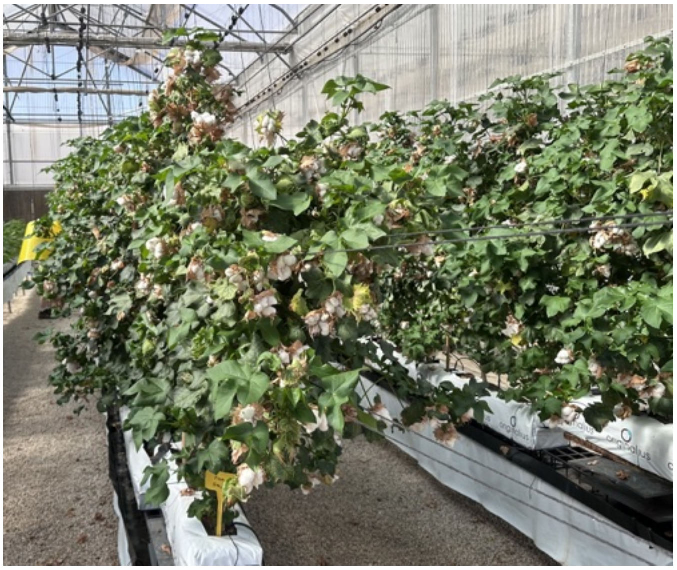


Fig. 1: Greenhouse cotton plantation at “Centro de Edafología y Biología Aplicada del Segura” (CEBAS), CSIC.

2) Use data augmentation to improve prediction accuracy.
3) Train an object detection model with the augmented dataset capable of discerning both ready-to-harvest and non-ready-to-harvest cotton.
4) Achieve a model with sufficient inference speed for real-time tracking and counting of cotton bolls for rapid crop monitoring.

III. State of the Art

Recent advancements in robotic harvesting systems have significantly enhanced agricultural productivity, particularly in the cotton industry. Robotic systems equipped with advanced vision and navigation technologies have been developed to improve the efficiency and precision of harvesting operations. A comprehensive review by Droukas et al. [9] highlights various robotic harvesting systems, emphasizing the integration of vision systems, motion planning, and end-effector designs tailored for different crops. These systems typically consist of a mobile platform, a robotic arm, and various sensory and navigation technologies to facilitate accurate crop detection and harvesting.

Regarding perception, object detection and classification in agriculture have gained significant attention recently, as the sector increasingly recognizes the potential for these technologies to be applied across all phases of the agricultural process – from harvesting to supply, including quality control and labelling.

Computer vision models have more than doubled in accuracy over the past four years [10], enabling the implementation of accurate, flexible, and powerful object detectors. For instance, Liu et al. [11] developed a YOLOX variation to detect and count small unopened cotton bolls, achieving an mAP of 92.75%, surpassing the original model YOLOv3∼v5 [12].

Semantic segmentation methods offer an alternative to object detection. Singh et al. [13] proposed a triplet of custom models that slightly outperform other segmentation models, though at the cost of increased computation time. Lv and Wang [14] focused on cotton growth phases and presented an optimized PSPNet [15] for segmenting and classifying cotton bolls, making it particularly relevant to our research.

Finally, if we focus on other implementations of autonomous harvesters, Similar object detection methods have been applied to other crops, often utilizing generic models trained on custom datasets. Wang et al. [16] proposed an implementation of YOLOv8 to detect strawberry ripeness, achieving 97.81% accuracy and 96.36% recall on their custom dataset (1187 images). Lenz et al. [17] also employed YOLOv8 to detect “pepper” and its “peduncle”, with good results in object detection and tracking. Yoshida et al. [18] used SSD for fruit detection, exceeding 95%.

While much work has been done on computer vision for crop harvesting, research specifically focusing on cotton boll harvesting appears less extensive. Nonetheless, the trend across crops is to utilize existing, well-established models and train them on custom datasets.

IV. Dataset Construction

We needed a computer vision model trained on data that reflects those specific conditions to ensure accurate detection of cotton bolls within our greenhouse environment. Due to the lack of suitable public datasets, we have created a custom-labelled dataset, in collaboration with CEBAS-CSIC, comprising both ready-to-harvest (ripe) and non-ready-to-harvest (unripe) cotton bolls.

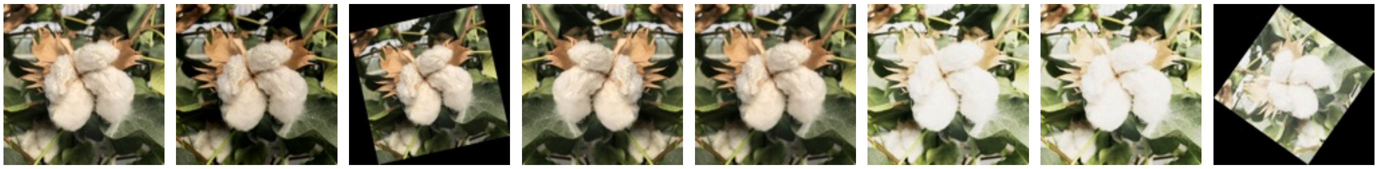
Images were collected from the CEBAS cotton plantation throughout its life cycle, with cultivar Intercott-211. This focus is crucial, as accurately distinguishing between mature and developing cotton is essential for harvesting the highest quality fibres.

The cotton bolls are harvested when they are fully open and dry. The husk that encloses the cotton and seeds splits into the different lobes of the boll and dries out. This transition is visually confirmed as the husk changes colour from green, indicating an immature stage, to brown, indicating maturity. This colour change and drying process serve as the primary parameters to determine the readiness for harvest. The decision was not solely based on experience but rather on these observable physiological indicators.

With the given information, two distinct classes for the model to train and predict were defined:

- Class “Unripe Cotton”: This class encompasses cotton buds, flowers, closed bolls, and open bolls that are yet not ready for harvest.
- Class “Ripe Cotton”: This class includes only cotton bolls ready for harvest.

The resulting dataset, named “Ready-To-Harvest Cotton Discerning Imageset (RTH-CONDIS)”, comprises 409 images



(OG) (B) (R) (H) (BL) (MN) (COMB0) (COMB1)

Fig. 2: Data augmentation techniques over a picture of a ready-to-harvest cotton boll. Refer to Table I for term meanings.

captured at a resolution of 15MP (5184x3024 pixels)[1]. Photos include diverse samples of both ripe and unripe cotton, captured at varying distances, mostly close-up ones (at a distance of approximately 50cm). All pictures were taken at the same time and under similar weather conditions, in the morning between 10 am and 12 pm, during June and July (summer). Consequently, variables like light conditions, shadows or seasonal variation are controlled.

Pictures were taken inside a 500m^2 polycarbonate greenhouse. The maximum temperatures reached 32ºC during the day and 20ºC at night, with relative humidity levels of 50% during the day and 85% at night. No shading net was used when the photos were taken.

Finally, LabelImg [19] was used to meticulously annotate each cotton boll individually, ensuring accurate labelling for subsequent model training.

V. Object detection Model

For the binary classification, we need a flexible framework. By looking at previous work in related fields [20], we see that YOLO provides a very powerful computer vision model, suitable for object detection, binary classification and multiple class classification, among others.

In the comparative graphic presented by [20], we can see that the different versions of YOLOv8 provide the best results. There is also an improvement between YOLOv8 models and previous YOLO versions, like YOLOv5 or YOLOv3. We will test and use YOLOv8, in its different sizes, as a computer vision model for this study. In addition, the solution is planned to be suitable for mobile, low-resource systems, so prediction time and memory usage are also core metrics.

YOLOv8 [21] is available in different models, depending on the number of parameters available. In increasing order of number of parameters, and decreasing order of image prediction frequency: YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, YOLOv8x.

Choosing a model that provides a good mAP is decisive. As versions n, s and m can provide predictions in real-time with decent mAP estimations, versions l and x were discarded for their model's parameter size, GPU memory usage, and lower frequency of predictions, compared to the other models.

VI. Experiments

Final metrics depend on the choice of the YOLOv8 version. This will directly impact the inference speed performance, as the final model will be deployed on an autonomous robot with limited computational resources. Therefore, we have evaluated both object detection and classification performances on every model.

In addition, data augmentation is proposed as a solution to enhance the dataset and make it more robust against over-fitting and ambient condition changes. Yang et al. [22] presented a survey on different techniques of data augmentation for image datasets and different computer vision tasks, concluding that it is needed to perform dataset augmentation when datasets are reduced, specific or tend to overfit. For this purpose, we will use the Albumentations library [23], allowing us to generate new images from rotations, brightness value changes, and other techniques in the YOLO labelling format.

When selecting properties to modify for data augmentation, it is important to prioritize those that best contribute to the model's learning process while remaining plausible in real-life scenarios. To address the specific challenges of this dataset, the following properties will be modified: brightness, rotation, horizontal flipping, blurring, and multiplicative noise (Fig. 2). In addition, more than one property can be changed at a time. Therefore, based on the improvement of each data augmentation property alone, we will also test the combination of the two most effective individual properties (COMB0) and the combination of all of them (COMB1) [24]. Our goal is to see which of these techniques, makes the trained model more robust in its ability to generalize to new, unseen data. By systematically exploring these data augmentation techniques, we aim to identify the optimal strategy for improving the model's performance and generalization capabilities.

Experimental procedures will be executed as follows:

1) Evaluate each data augmentation technique (single properties, best two properties and all together) in YOLOv8s[2].
2) Select the most effective augmentation technique based on model performance metrics.
3) Evaluate the augmented dataset for each version of YOLOv8.
4) Select a YOLOv8 version based on validation metrics.

Ultralytics library has been used for training, testing and predicting. YOLOv8 is not trained with any other dataset. It starts from an empty model and is trained with RTH-CONDIS.

[1] RTH-CONDIS is publicly available using the following DOI reference: https://doi.org/10.6084/m9.figshare.26214737

[2] YOLOv8s was chosen for its medium parameter size and computational training cost.

TABLE II: Comparative between models with data augmentation on YOLOv8s

TAD	precision	recall	F$_1$	mAP50	mAP50-95
OG	0.844	0.801	0,822	0.838	0.510
B	0.832	0.827	0.830	0.889	0.570
R	0.866	0.836	0.851	**0.910**	0.586
H	0.850	**0.863**	0.856	0.887	0.565
BL	0.913	0.856	**0.884**	0.898	0.553
MN	**0.946**	0.803	0.869	0.895	0.553
COMB0	0.911	0.804	0.854	0.891	0.568
COMB1	0.901	0.852	0.876	0.902	**0.590**

VII. Results

A. Dataset augmentation analysis

After applying the previously mentioned techniques, table II shows the performance metrics using YOLOv8s. We found that, due to their high precision value and mAP50, blurring (BL) and multiplicative noise (MN) provide the best results. For the combination of techniques, we tested blurring and multiplicative noise together (COMB0), as these individually yielded the best overall performance metrics. We also tested all techniques applied simultaneously (COMB1), as previous image recognition experiments have shown that combined methods often outperform single methods [24]. Results indicate that COMB0 does not yield significantly improved metrics, whereas COMB1 shows an overall high score across all of them.

Taking into account that precision and recall are important parameters in this study and that the mAP50 value does not vary very much, COMB1 has been selected as the combination of data augmentation techniques to be applied in the RTH-CONDIS training set.

Comparing the original dataset with the augmented one (COMB1) reveals significant improvement in model metrics: precision and recall metrics share an approximate increment of 0.05; mAP50 increases from 0.838 to 0.902, a difference of 0.064; mAP50$-$95 increases from 0.510 to 0.590, a difference of 0.08.

Dataset partition into training, testing and validation sets, with percentages stated (80% for training, 10% testing, 10% validation) is done using randomness.

B. Model performance

This section evaluates the performance of different YOLOv8 models trained on our augmented RTH-CONDIS dataset. Table IV provides a comparative overview of the model metrics, and Figure 3 shows examples of some model predictions. As mentioned in Section V, due to the real-time requirements of our task, we focus on evaluating the n, s and m versions of YOLOv8.

First of all, in terms of speed both CPU and GPU performances have been compared for the three sizes of models, as presented in Table III[3], all suitable for real-time detection.

TABLE III: Comparative between inference speed between models over GPU and CPU executions

YOLOv8 version	Device	Inference Time (ms/image)	FPS
YOLOv8n	CPU	387.9	2.6
	GPU	5.2	192.3
YOLOv8s	CPU	544.6	1.8
	GPU	8.8	113.6
YOLOv8m	CPU	517.3	1.9
	GPU	14.1	70.9

TABLE IV: Comparison between model sizes for augmented dataset training

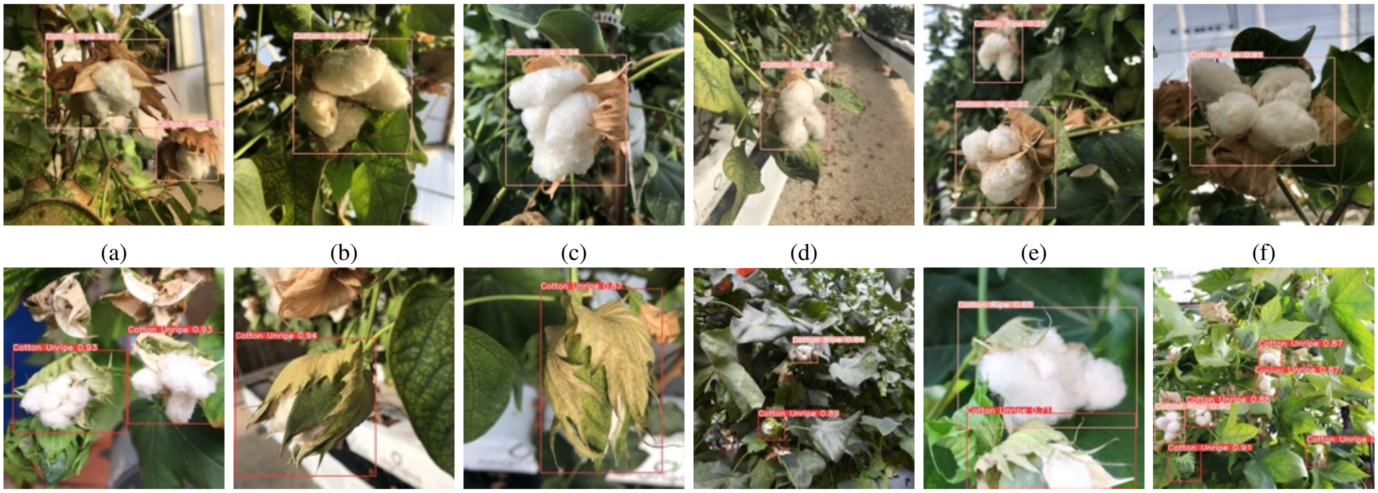
Model	epochs	precision	recall	mAP50	mAP50-95
YOLOv8n	385	**0.920**	0.807	**0.902**	**0.585**
YOLOv8s	272	0.901	**0.852**	**0.902**	**0.585**
YOLOv8m	216	0.815	0.831	0.884	0.582

All three versions of the model demonstrated high detection performance. YOLOv8n and YOLOv8s achieved more than 0.9 mAP50 score, while YOLOv8m had a 0.884 mAP50. We also observed an inverse relationship between the number of epochs required and the model size. The YOLOv8n version took the most epochs to find a minimum, while YOLOv8m took the least. These findings highlight the trade-off between model size, training time (epochs), and performance, providing valuable insights for selecting the optimal YOLOv8 version for our real-time task. In the particular task of detecting, tracking, and counting cotton bolls, *precision* and *recall* are equally important. Therefore, the model with the most balanced performance will be the one selected. As shown in Table IV, both the n and s models demonstrate the best overall object detection metrics. However, the s version exhibits the most balanced *precision* versus *recall* performance, as it can be seen when comparing F1 curves in Fig. 4.

Regarding predictions, in Figure 3 we see that the model detects better when gets closer to the cotton bolls. RTH-CONDIS is made of mainly close-up pictures of cotton, so when obstacles are found in the way, or distance increases, the detection and classification precision decreases proportionally. This project is thought for robot implementations with cameras close to the crops. In the future, when the object detection model is included in the harvesting mobile robot, we will only collect ripe cotton with high confidence. This criterion is substantiated by Zhang et al. [6] work, showing that ready-to-harvest cotton bolls stay in this phase long without losing quality. Harvesting the remaining cotton bolls from the previous iteration in a new one is preferable, to collecting not ready-to-harvest cotton.

If we analyse the confusion matrix at Fig. 5, we observe prediction values for the “Cotton ripe” class of 0.9, while the “Cotton unripe” class has a lower prediction rate, with a value of 0.81. Another notable aspect within the matrix is the value “Cotton unripe/background”, which denotes that the model had predicted unripe cotton when it was the image’s

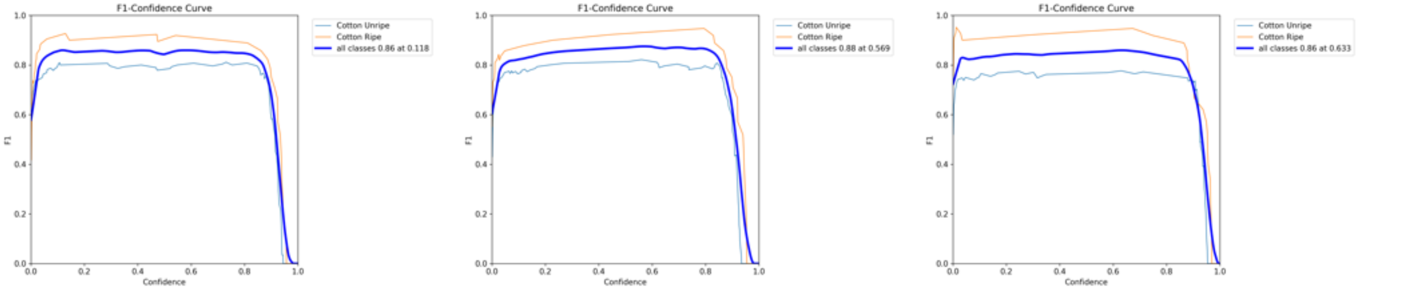
[3] For testing speed performance: GPU model = Nvidia GTX 1070Ti 8GB, CPU model = Intel Core i5 9600K, 16GB of RAM



(a) (b) (c) (d) (e) (f)

(g) (h) (i) (j) (k) (l)

Fig. 3: Examples of cotton detection with RTH-CONDIS-trained YOLOv8s. a-f: ready-to-harvest cotton boll detection. Bottom row: g-i non-ready-to-harvest cotton bolls detected; j-l mixed cases (ready and non-ready-to-harvest cotton bolls). The model is optimal for close detection and classification and decreases its predictions when the distance to the bolls is incremented.



(a) YOLOv8n (b) YOLOv8s (c) YOLOv8m

Fig. 4: Comparison between F1 curves for the three evaluated versions of YOLOv8. Observe how the YOLOv8s model achieves the highest F1 score (0.88) at a confidence level of 0.569. This high F1 score indicates an optimal balance between precision and recall, meaning that it effectively minimizes both false positives and false negatives. While YOLOv8s achieves the highest F1 score, YOLOv8n and YOLOv8m demonstrate slightly lower scores (both 0.86), indicating a potential trade-off between speed and accuracy. Ultimately, the choice of the most suitable model depends on the specific requirements of the application and the relative importance of precision, recall and inference speed.

background. This event occurs in 11 cases, of which 10 are predicted as "Cotton Unripe" and 1 is predicted as "Cotton Ripe". Compared to the total number of labels in the validation set, the number of confusions in the prediction for the class "background" is negligible. This appears because YOLO applies this "class" to identify the cases where the model detected something when the background was misclassified with some cotton boll (the ground truth is the background and the predicted label is some class, and vice versa).

VIII. Conclusions and Future Work

In this paper, we propose a YOLOv8 model solution to accurately and comprehensively identify the ripening phases of cotton bolls in complex environments like a greenhouse. After analysing the training/validation metrics, we conclude that the best model version for our requirements is the YOLOv8s model. Although YOLOv8n has the best precision, its recall is too low compared to the other versions. YOLOv8s provides the best overall precision and recall metrics combined, the best mAP50 and the best $mAP50-95$ metrics. Moreover, its prediction time is suitable for any real-time detectors.

Despite having a limited dataset, compared to other similar works, our model achieves very good detection and classification performances, while keeping a low inference time. Larger versions of YOLOv8 couldn't be tested (l,x) because of GPU memory capacity, but since this project is focused on a mobile robot system, they stay out of scope[4]. At the same time, it would be possible to achieve higher detection and classification rates with a more complex and bigger

[4] For training: GPU model = NVIDIA A10 (x2), CPU model = Intel(R) Xeon(R) Gold 6326, 128GB of RAM

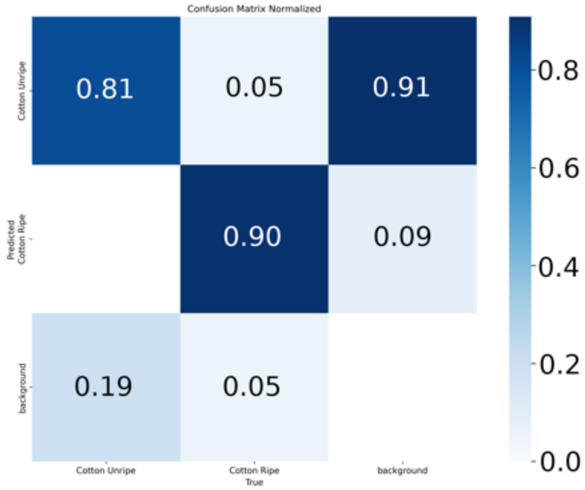


Fig. 5: Normalized confusion matrix for YOLOv8s with data augmentation.

dataset, since a richer cotton ripeness representation would be obtained.

For the next steps, we will increase the dataset with more diversity of samples and classes, so that we can recognize different stages of cotton growth (early cocoons, stages of flower growth, early cotton bolls, open cocoons not ready to harvest). Our goal is to perform semantic segmentation, obtain the predicted labels, compute the centroid, and localise the cotton boll centre via RGB-D cameras. This research is part of the DEMETER 5.0 project, where a robot has to recognise the growth stage, estimate the growth time needed for a cotton boll to be collected, and keep track of crops. Thus, the robot can keep a world state representation, optimise reasoning, and pick those cotton bolls that are ready-to-harvest.

Acknowledgment

This work has been partially funded by DEMETER 5.0 project PLEC2022-009289 funded by MCIU/ AEI /10.13039/501100011033 and by the "European Union NextGenerationEU/PRTR". We would also like to thank the personnel at CEBAS-CSIC for maintaining the greenhouse with the cotton plants, especially Cecilia Martín, who provided us with cotton knowledge and the pictures necessary to develop the dataset.

References

[1] G. Soley, "Cotton: World markets and trade," 2024.

[2] P. K. Mishra, A. Sharma, and A. Prakash, "Current research and development in cotton harvesters: A review with application to indian cotton production systems," *Heliyon*, vol. 9, no. 5, p. e16124, 2023.

[3] F. A. Nayra, L. M. Renildo, M. B. Cíntia, T. M. Myllena, L. C. William, and A. V. Carlos, "Mechanical harvest methods efficiency and its impacts on quality of narrow row cotton," *African Journal of Agricultural Research*, vol. 13, no. 41, pp. 2263–2268, 2018.

[4] H. Gharakhani, J. A. Thomasson, and Y. Lu, "An end-effector for robotic cotton harvesting," *Smart Agricultural Technology*, vol. 2, p. 100043, Dec. 2022.

[5] W. Wu, J. Wu, Z. Shen, L. Yin, and Q. Liu, "Research on an embedded system of cotton field patrol robot based on ai depth camera," in *Lecture Notes in Computer Science*, p. 529–538, Springer Nature Singapore, 2023.

[6] X. Zhang, Q. Yang, R. Zhou, J. Zheng, Y. Feng, B. Zhang, Y. Jia, X. Du, A. Khan, and Z. Zhang, "Perennial cotton ratoon cultivation: A sustainable method for cotton production and breeding," *Front. Plant Sci.*, vol. 13, p. 882610, 2022.

[7] X. Zhang, X. J. Kong, R. Y. Zhou, Z. Y. Zhang, J. B. Zhang, L. S. Wang, and Q. Wang, "Harnessing perennial and indeterminant growth habits for ratoon cotton (gossypium spp.) cropping," *Ecosyst Health Sust*, vol. 6, p. 1715264, 2020.

[8] S. Thapa, G. C. Rains, W. M. Porter, G. Lu, X. Wang, C. Mwitta, and S. S. Virk, "Robotic multi-boll cotton harvester system integration and performance evaluation," *AgriEngineering*, vol. 6, no. 1, pp. 803–822, 2024.

[9] L. Droukas, Z. Doulgeri, N. L. Tsakiridis, D. Triantafyllou, I. Kleitsiotis, I. Mariolis, D. Giakoumis, D. Tzovaras, D. Kateris, and D. Bochtis, "A survey of robotic harvesting systems and enabling technologies," *Journal of Intelligent amp; Robotic Systems*, vol. 107, Jan. 2023.

[10] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *Proceedings of the IEEE*, vol. 111, no. 3, pp. 257–276, 2023.

[11] Q. Liu, Y. Zhang, and G. Yang, "Small unopened cotton boll counting by detection with mrf-yolo in the wild," *Computers and Electronics in Agriculture*, vol. 204, p. 107576, 2023.

[12] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "Yolox: Exceeding yolo series in 2021," 2021.

[13] N. Singh, V. Tewari, P. Biswas, and L. Dhruw, "Lightweight convolutional neural network models for semantic segmentation of in-field cotton bolls," *Artificial Intelligence in Agriculture*, vol. 8, pp. 1–19, 2023.

[14] Q. Lv and H. Wang, "Cotton boll growth status recognition method under complex background based on semantic segmentation," in *2021 4th International Conference on Robotics, Control and Automation Engineering (RCAE)*, pp. 50–54, 2021.

[15] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[16] C. Wang, H. Wang, Q. Han, Z. Zhang, D. Kong, and X. Zou, "Strawberry detection and ripeness classification using yolov8+ model and image processing method," *Agriculture*, vol. 14, no. 5, 2024.

[17] C. Lenz, R. Menon, M. Schreiber, M. P. Jacob, S. Behnke, and M. Bennewitz, "Hortibot: An adaptive multi-arm system for robotic horticulture of sweet peppers," *arXiv preprint arXiv:2403.15306*, 2024.

[18] T. Yoshida, Y. Onishi, T. Kawahara, and T. Fukao, "Automated harvesting by a dual-arm fruit harvesting robot," *ROBOMECH Journal*, vol. 9, Sept. 2022.

[19] Tzutalin, "Labelimg." Git code, 2015.

[20] C. Wang, H. Wang, Q. Han, Z. Zhang, D. Kong, and X. Zou, "Strawberry detection and ripeness classification using yolov8+ model and image processing method," *Agriculture*, vol. 14, no. 5, p. 751, 2024.

[21] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics yolov8," 2023.

[22] S. Yang, W. Xiao, M. Zhang, S. Guo, J. Zhao, and F. Shen, "Image data augmentation for deep learning: A survey," *arXiv preprint arXiv:2204.08610*, 2022.

[23] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, "Albumentations: Fast and flexible image augmentations," *Information*, vol. 11, no. 2, 2020.

[24] P. Pawara, E. Okafor, L. Schomaker, and M. Wiering, "Data augmentation for plant classification," *In Proc. International Conference on Advanced Concepts for Intelligent Vision Systems*, pp. 615—-626, 2017.