



## Original papers

## CottonSense: A high-throughput field phenotyping system for cotton fruit segmentation and enumeration on edge devices



Farshad Bolouri<sup>a,\*</sup>, Yildirim Kocoglu<sup>b</sup>, Irish Lorraine B Pabuayon<sup>c</sup>, Glen Lorin Ritchie<sup>c</sup>, Hamed Sari-Sarraf<sup>a</sup>

<sup>a</sup> Department of Electrical and Computer Engineering, Edward E. Whitacre Jr. College of Engineering, Texas Tech University, Lubbock, TX, United States

<sup>b</sup> Bob L. Herd Department of Petroleum Engineering, Edward E. Whitacre Jr. College of Engineering, Texas Tech University, Lubbock, TX, United States

<sup>c</sup> Department of Plant and Soil Science, Davis College of Agricultural Sciences & Natural Resources, Texas Tech University, Lubbock, TX, United States

## ARTICLE INFO

## Keywords:

Cotton  
High-throughput Phenotyping  
Fruit Counting  
Deep learning  
Segmentation  
Edge Device

## ABSTRACT

High-throughput phenotyping (HTP) has become a powerful tool for gaining insights into the genetic and environmental factors that affect cotton (*Gossypium* spp.) growth and yield. With the recent advances in the field of computer vision, namely the integration of deep learning algorithms, the accuracy and efficiency of HTP systems have improved dramatically, enabling them to automatically quantify such fundamental phenotypic traits as fruit identification and enumeration. However, there is currently no HTP system available for counting all the reproductive phases of cotton crop that can be deployed in agronomic field conditions throughout the growing season. This study presents CottonSense, an advanced HTP system that overcomes the challenges of deployment across multiple growth periods by effectively segmenting and enumerating cotton fruits at four stages of growth, including square, flower, closed boll, and open boll. Consequently, CottonSense enhances agronomic management through increased opportunities for data collection and analysis. Using RGB-D cameras, it captures and processes both two and three-dimensional data, facilitating a wider range of phenotypic trait extractions such as crop biomass and plant architecture. To segment the cotton fruits, a Mask-RCNN model is trained and optimized for faster inference using TensorRT. The model yields an average AP score of 79% in segmentation across the four fruit categories. Moreover, the model's accuracy in estimating total fruit count per image is validated by a strong agreement with the counts given by ten domain experts, as reflected by an  $R^2$  value of 0.94. Furthermore, to accurately count the segmented fruits over large populations of plants, an enumeration algorithm based on a tracking strategy is developed that achieves an  $R^2$  value of 0.93 when compared to hand-counted fruits in the field. The proposed HTP system, which is implemented entirely on an edge computing device, is cost-effective and power-efficient, making it an effective tool for high-yield cotton breeding and crop improvement. The code for CottonSense is publicly available at <https://github.com/FeriBolour/CottonSense>.

## 1. Introduction

Cotton (*Gossypium* spp.) accounts for nearly 80 % of the global natural fiber production, with China, India, the United States, and Pakistan being the top producers (Fangueiro and Rana, 2016). Cotton yield is a function of several physiological factors, with the number and size of individual fruits on the plant being the fundamental determinants. Therefore, the number of fruits is a key indicator of the potential yield of a field and is also useful in understanding the growth conditions of the crop. This information can be used by cotton growers to both make management decisions and determine the impact of environmental

conditions, pressure from weeds and insects, and management on crop yield (Adke et al., 2022; Pabuayon et al., 2021). However, counting cotton fruits by hand in the field is a laborious task that is usually limited to a few plants within a field or field plot (Ritchie et al., 2012). Traditional methods of yield prediction based on manual sampling (Huang et al., 2016) or visual inspection and experience (Feng et al., 2019) are prone to errors and are not practical for evaluating large numbers of plots commonly encountered in plant breeding programs.

To address these issues, researchers have developed non-invasive, high-throughput phenotyping (HTP) techniques that automate plant trait measurements and thereby reduce time and labor requirements

\* Corresponding author at: Texas Tech University, ECE department, room 121, 1012 Boston Ave, Lubbock, TX 79409, United States.

E-mail address: [farshad.bolouri@ttu.edu](mailto:farshad.bolouri@ttu.edu) (F. Bolouri).

(Normanly, 2012; Pabuayon et al., 2019). Such techniques include computer vision-based systems combined with agricultural robots (Oberti and Shapiro, 2016) and unmanned aerial vehicles (Zaman-Allah et al., 2015) that aim to automate fruit enumeration and yield prediction, increase efficiency, and reduce human error (Qureshi et al., 2017; Rahnemoonfar and Sheppard, 2017; Sun et al., 2019). However, additional advancements are required for these methods to be effective in indeterminate crops such as cotton which produce fruit over long periods of time and have several fruits at differing stages at the same time. These advancements include the ability of sensing systems to collect measurements in agronomic field situations and the capability of the systems to economically and accurately identify and discriminate among fruit of several stages simultaneously over large areas of land (Pabuayon et al., 2021; Snowden et al., 2013).

This paper presents CottonSense, an HTP system for cotton plants that bridges the gaps inherent to traditional methods and advances the state-of-the-art established through recent automations by making four novel and significant contributions:

- I. **Data collection in standard agronomic field conditions:** Yields in cotton and other crops are dependent on management strategies, making it essential to test cultivars in agronomic fields with similar management to that employed in commercial production. CottonSense is capable of collecting measurements in agronomic cotton fields under the same conditions (e.g., plant density and spacing) as those used in commercial settings.
- II. **Two and three-dimensional fruit identification in four stages of development:** It is important for an HTP system to capture the fruiting dynamics throughout the growing season, as it allows for more opportunities for agronomic management (Constable and Bangs, 2015). This can be achieved by using two-dimensional (2D) computer vision systems to detect different growth stages of the fruits. Additionally, by utilizing three-dimensional (3D) remote sensing and computer vision, a wider range of phenotypic traits can be extracted, including crop biomass estimation, plant architectural trait analysis, and digital elevation model generation (Wallace et al., 2017; Wang et al., 2016). CottonSense is capable of identifying and discriminating cotton fruits in the square, flower (bloom), closed boll (green/unopened boll), and open boll stages in both 2D and 3D. This involves not only identifying the presence and location of the fruits (fruit detection) but also all the pixels that compose them, from which one can get the fruits' boundaries and separate them from the stems and leaves (fruit segmentation).
- III. **Enumeration of identified fruits over a large population of plants:** Effective crop monitoring and yield estimation dictates that the counting capability of existing HTP systems be extended to a larger population of plants (Ritchie et al., 2012). This study demonstrates an enumeration technique that utilizes a fruit-tracking strategy across consecutive video frames to cover a sufficiently large plant population.
- IV. **Cost and power efficiency:** Unlike any of the existing HTP approaches, CottonSense utilizes an edge computing device (Shi et al., 2016), the NVIDIA Jetson AGX Xavier (NVIDIA, 2019), for data acquisition and local processing, resulting in a cost-effective and power-efficient solution. Additionally, the identification subsystem employs TensorRT (NVIDIA, 2016), a deep learning inference library, that optimizes the computations of neural networks and brings the processing speed closer to that of workstation GPUs.

Recently, several studies have explored the use of cotton plant HTP for fruit enumeration (Table 1). For instance, Sun et al. (2019) proposed novel image processing algorithms for 2D open boll segmentation using their color and spatial features and then estimating their count using geometric-feature-based methods. The same group proposed an

**Table 1**  
Summary of state-of-the-art cotton plant HTP studies and their comparison to the proposed system.

Agronomic Field Conditions	2D Fruit Detection			2D Fruit Segmentation			Large-Scale Fruit Enumeration			3D Data Collection			3D Fruit Segmentation			
	Open Boll	Closed Boll	Flower	Square	Open Boll	Closed Boll	Flower	Square	Open Boll	Closed Boll	Flower	Square	Open Boll	Closed Boll	Flower	
Adle et al. (2022)	✓															
Sun et al. (2022)	✓															
Yan Zhang et al. (2022)	✓															
Jiang et al. (2020)																
Sun et al. (2020)																
Sun et al. (2019)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
CottonSense	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

alternative approach using 3D photogrammetric mapping of the cotton rows, followed by point cloud segmentation and clustering techniques to individualize and count the open bolls (Sun et al., 2020). Although these systems showcase similar characteristics as (II) and (III), the enumeration of open bolls that contrast with vegetation does not require specific computer decision-making and limits measurement opportunities to the end of the growth cycle, thereby constraining agronomic management options. Additionally, the data collection systems in these studies, have not shown to be operational in agronomic field conditions (I), as they were only tested in research fields with wider than normal row spacings, which range from 0.9 to 1.0 m width in most systems throughout the US cotton growing region.

HTP researchers have also harnessed the recent transformative advances in deep learning to address the automatic 2D detection of cotton fruits at different stages of growth. Faster-RCNN (Ren et al., 2017) was used to identify emerging blooms (Jiang et al., 2020), custom object detection networks were proposed for closed boll localization (Sun et al., 2022; Yan Zhang et al., 2022), and weakly supervised models were explored to count open bolls (Adke et al., 2022). Despite these noteworthy advances, as seen in Table 1, none of these studies offer a comprehensive solution (II and III), that covers all four growth stages for a large plant population.

In summary, this study introduces a system for cotton fruit identification and enumeration that addresses the limitations of previous systems. CottonSense is capable of collecting measurements and identifying the cotton fruits in four stages of their growth cycle in 2D and 3D under agronomic field conditions. Additionally, it stands out as the only system that addresses cost and power efficiency (IV) by running entirely on an edge computing device.

## 2. Materials and methods

### 2.1. Experimental field setup and data collection

The experimental field was located at BASF Agricultural Solutions Breeding & Trait Development Station in Lubbock, TX, USA ( $33^{\circ} 35' 01.0''$  N,  $101^{\circ} 46' 35.6''$  W). Data were collected from a larger field trial with 22 cotton cultivars that included commercial and experimental lines, replicated four times in a randomized complete block design within each of four irrigation treatments based on evapotranspiration (ET) calculations from a weather station (non-irrigated; 30 % ET; 60 % ET; and 90 % ET). The plots were 7.6 m long with four rows per plot and a 1.0 m row spacing. Data were collected from July 2021 to November 2021 using two Intel RealSense D435i depth cameras (Intel, 2023) mounted on a tractor platform. The limited spacing between the rows often required the use of two cameras in order to capture the entire height of the plants. As seen in Fig. 1, the cameras were positioned on one side of the plants perpendicular to the ground and directly facing the plants. Controlling the cameras and storing and processing the data was

accomplished using an NVIDIA Jetson AGX Xavier (NVIDIA, 2019), which was mounted inside the tractor cabin (Fig. 2).

The data collected from the cameras included RGB-D images (registered RGB and depth images) and inertial measurement unit (IMU) information during the tractor's movement. The experiments were conducted both in daylight and at night using two LED light bars positioned on either side of the cameras (Fig. 2) to provide uniform illumination. In later versions of the acquisition system shown in Fig. 2, a mounting was built that used a shroud to minimize wind effects, color saturation from direct sunlight, and interference from plants in the adjacent rows. The system diagram presented in Fig. 3, provides an overview of the key processes for data collection and processing in CottonSense with section numbers placed next to the relevant components.

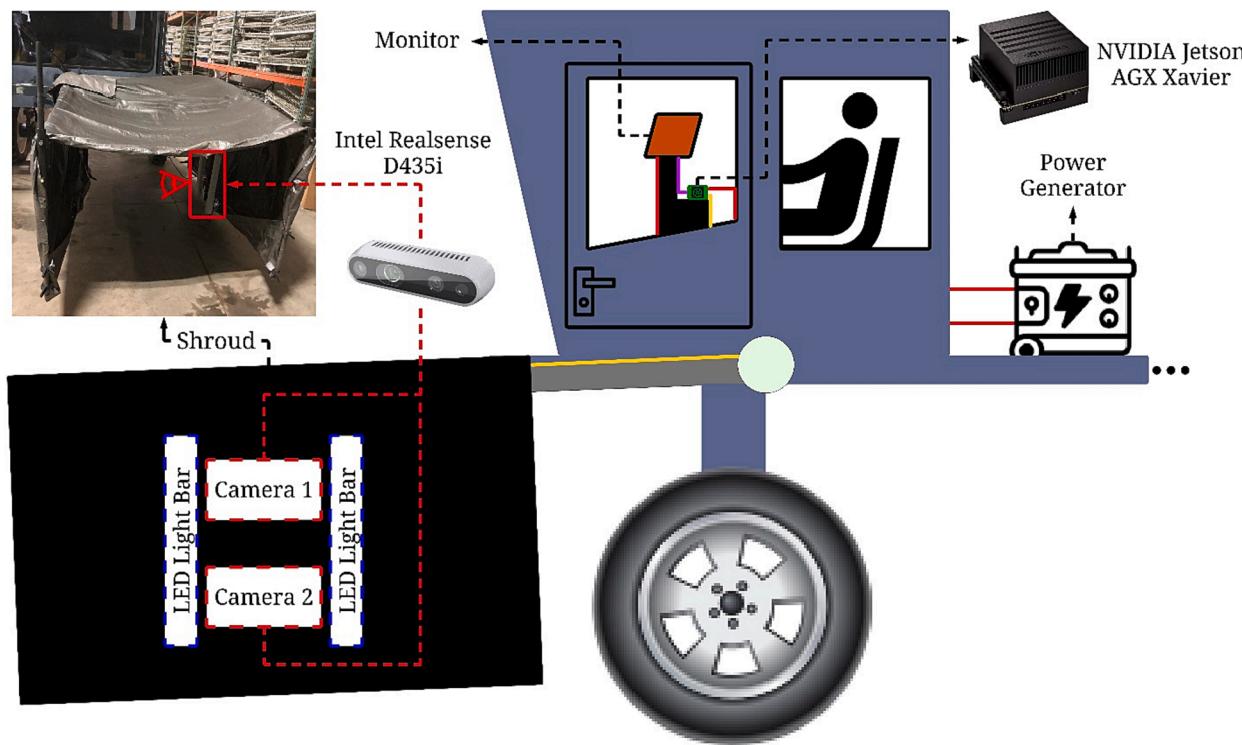
During the imaging of a row, a 3D point cloud was also constructed using the RTAB-Map library (Fig. 4). RTAB-Map (Labbé and Michaud, 2019) is an open-source library that provides the necessary tools and algorithms for utilizing measurements from various sensors such as RGB-D cameras and Lidar sensors to map the environment in 3D space. To produce a single point cloud of the row, RTAB-Map was configured to use the following process for each camera:

- I. Identifying visual features in RGB image frames using the default feature detection and descriptor algorithms; namely, GFTT (Shi and Tomasi, 1994) and Rotated BRIEF (Rublee et al., 2011), respectively.
- II. Computing affine transformation matrices for aligning and stitching the consecutive image frames by finding their corresponding detected features from (I) using the feature matching algorithm, FLANN (Muja and Lowe, 2009); see Fig. 5.
- III. Creating a 3D point cloud from each RGB-D image by associating a depth value to each pixel in the 2D RGB image, resulting in a set of points with both 3D position and color information.
- IV. Integrating the calculated transformation matrices from (II) with the IMU data obtained by the cameras to align and stitch corresponding point clouds of each image (III), resulting in a unified point cloud of the entire row. The IMU data enhances the point cloud stitching process by providing supplementary information about the camera's orientation and motion.

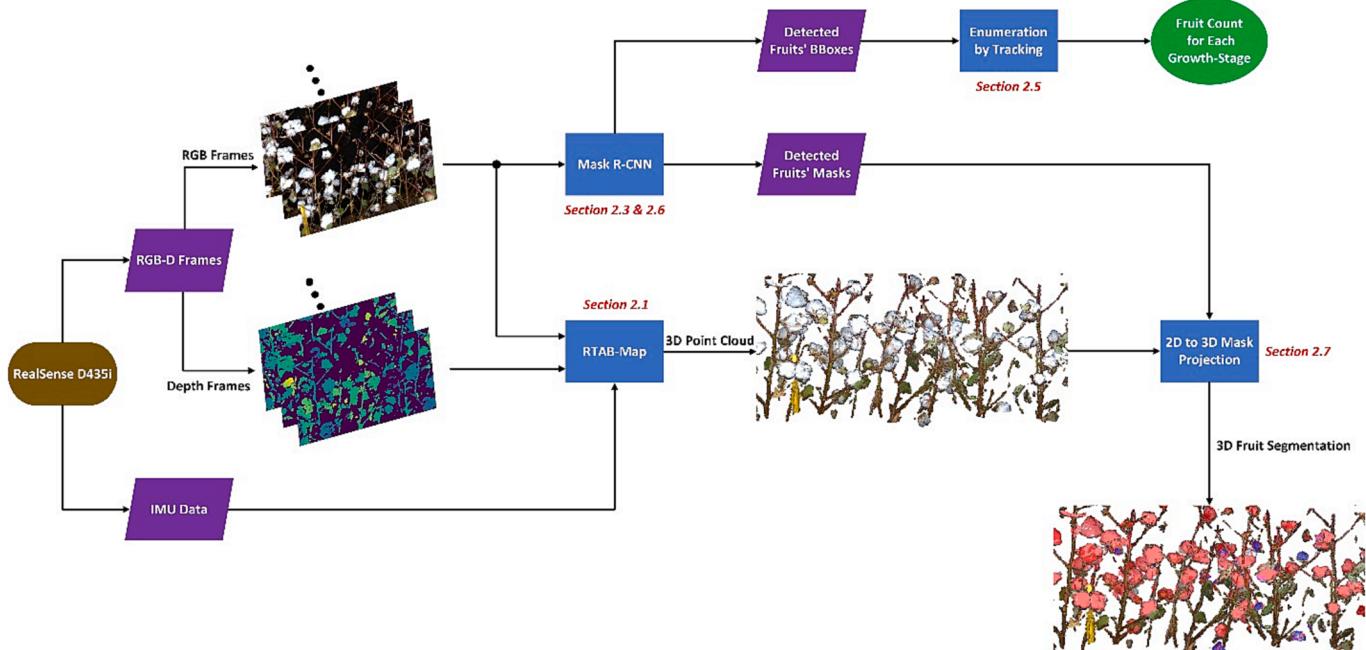
Lastly, to obtain a single point cloud of the row that captures the full height of the plants, it is necessary to stitch together the point clouds generated from each camera. To that end, a variant of the popular Iterative Closest Point (ICP) algorithm (Besl and McKay, 1992), called the Colored Point Cloud Registration (Park et al., 2017), was employed. The authors demonstrate that this method offers higher accuracy and robustness than ICP for colored point clouds. The example in Fig. 6 demonstrates the effectiveness of the Colored Point Cloud Registration algorithm in stitching the two point clouds.



**Fig. 1.** A) Cameras' position (in the red box) with respect to the plants – B) Imaging in the daylight – C) Imaging at Night.



**Fig. 2.** Data Collection Platform. This figure showcases the essential components of the data collection platform and their placement on the tractor.



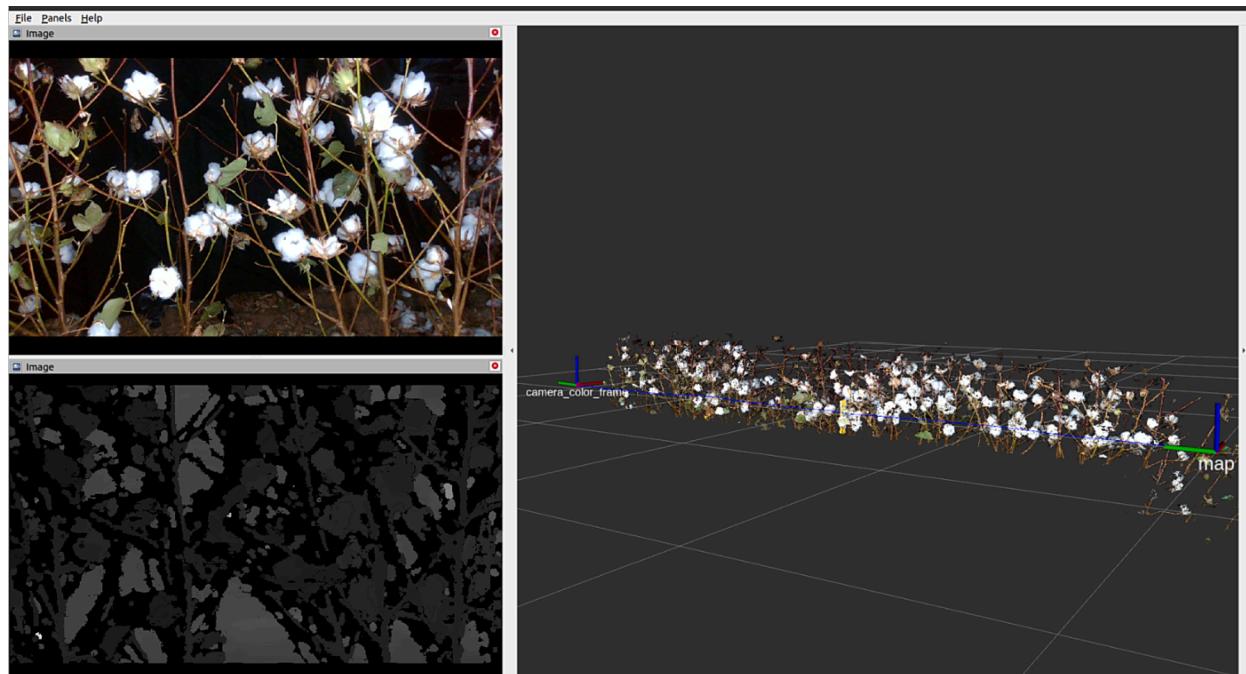
**Fig. 3.** CottonSense System Diagram. This figure illustrates the proposed phenotyping system, highlighting key processes and their corresponding section numbers.

## 2.2. Dataset generation for model development

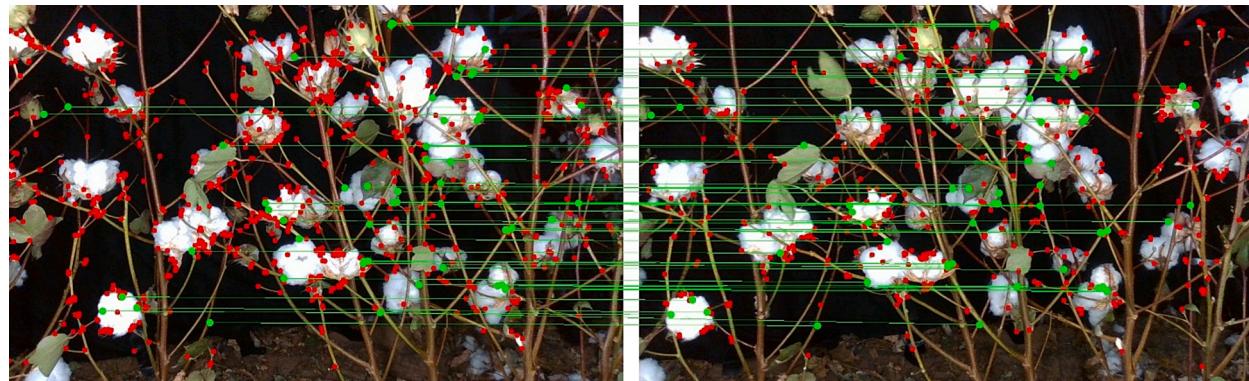
A multi-instance semantic segmentation architecture, details of which are discussed in the next section, was trained in a supervised mode to identify the cotton plant fruit in four developmental stages. It is well established in the machine learning community that regardless of the specific model choice, the training phase is impacted the most by the integrity and representativeness of the training data. In keeping with this, a considerable amount of resources and planning went into insuring

that the generated training dataset is accurate, consistent, and complete. To prepare the collected dataset for the development of the model, a 4-class labeling strategy was implemented. The classes, namely Square, Flower, ClosedBoll, and OpenBoll is presented in Fig. 7.

A team of 5 annotators contributed to the labeling process using MATLAB's Image Labeler tool (MathWorks, 2018), with each annotator assigned a unique set of images to label. The annotators identified the boundary of each fruit using polygon annotations provided by the tool. To ensure consistency and accuracy across all annotations, the following



**Fig. 4.** Snapshot of RTAB-Map for one of the cameras. This figure depicts the mapping being done in real-time for point cloud generation of a row. Top-left panel shows the RGB images, bottom-left panel shows the depth images, and right panel shows the point cloud of the row being constructed.



**Fig. 5.** Example of two consecutive RGB image frames used for the point cloud generation process. Green circles and lines depict the matched features detected and matched by the GFTT and FLANN algorithms, respectively. The red circles indicate detected features that did not match with a corresponding feature after the feature-matching process.

protocol was established and followed throughout the labeling process:

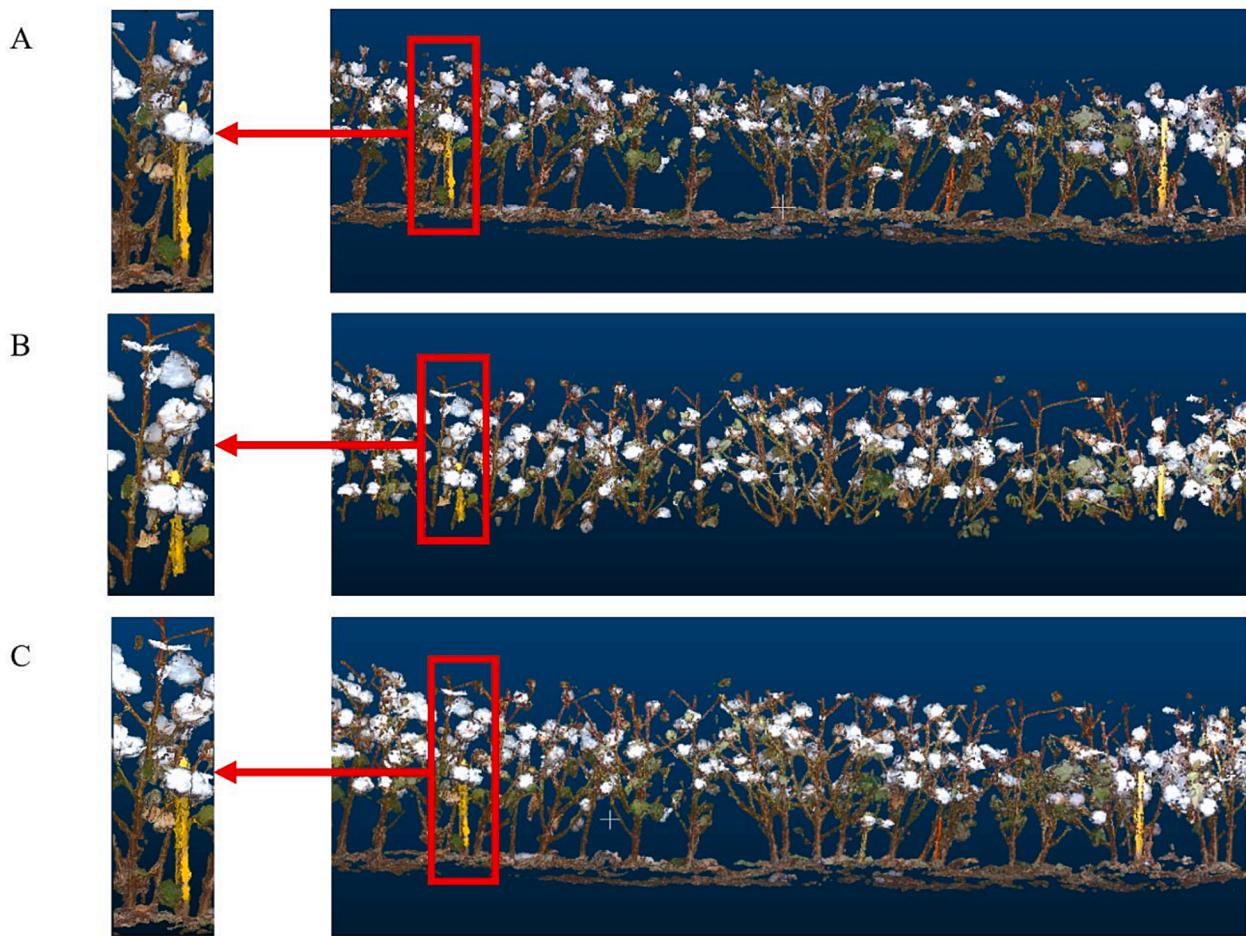
- I. When a fruit was found to be in a transitional stage (e.g., from Square to Flower), it was annotated and labeled as the more advanced of the two stages; see Fig. 8A.
- II. If a fruit was obstructed by other organs, a single polygon that enclosed all the fruit's visible parts was drawn instead of creating a separate polygon for each visible part; see Fig. 8B.
- III. In scenes with overcrowding of fruits and leaves, only the fruits with clearly identifiable boundaries were annotated, and the rest were left unannotated; see Fig. 8C.
- IV. Dropped fruits that are visible on the ground were not annotated as they no longer contribute to the plants' yield; see Fig. 8D.

After completing annotations for 184 images, a preliminary segmentation model (further elaborated in the next section) was trained to aid in both the model development process and the annotation of more images. The model was employed to generate initial annotations for the remaining images in the dataset. This expedited the annotation process

as annotators only needed to make minor adjustments to the automatically generated annotations, such as modifying labels or refining polygons. Additionally, in crowded scenes mentioned in (III), the model helped to identify the boundaries of certain fruits that were previously difficult to discern. As a result, the preliminary model not only saved time and effort for the annotators but also improved the quality and consistency of the final annotated dataset.

To ensure a balanced representation of the four categories for model development, 344 images were sampled temporally from the final annotated dataset. These images were then split into training and test sets with a ratio of 7:3 for each category; detailed in Table 2. As depicted in Figs. 9A and 9B, a lower number of images were sampled in the late-season than early to mid-season to avoid an overrepresentation of OpenBolls. This is due to the defoliation of plants before harvesting, which causes the cameras' field of view to no longer be obscured by dense foliage, allowing for the capture of more fruits. Fig. 9C confirms this, where the number of OpenBolls per image is notably higher than other categories during this period.

The distribution of the data used for model development is validated



**Fig. 6.** Example of point clouds generated and stitched from the 2 cameras. – A) Point cloud generated from the bottom camera. – B) Point cloud generated from the top camera. – C) Aligned and stitched point clouds using the Colored Point Cloud Registration algorithm.

by its coherence with the established literature on cotton fruit development and accumulation throughout the growing season (Fig. 10). As shown in Fig. 9D, it illustrates a consistent progression, where Squares dominate early on, followed by ClosedBolls in line with the flowering cycle, and then ClosedBolls are replaced with OpenBolls starting around 115–125 days after planting. Additionally, the dataset confirms that no further Flowers are observed after this point, and any Squares observed are guaranteed to be ClosedBolls.

### 2.3. Instance segmentation model

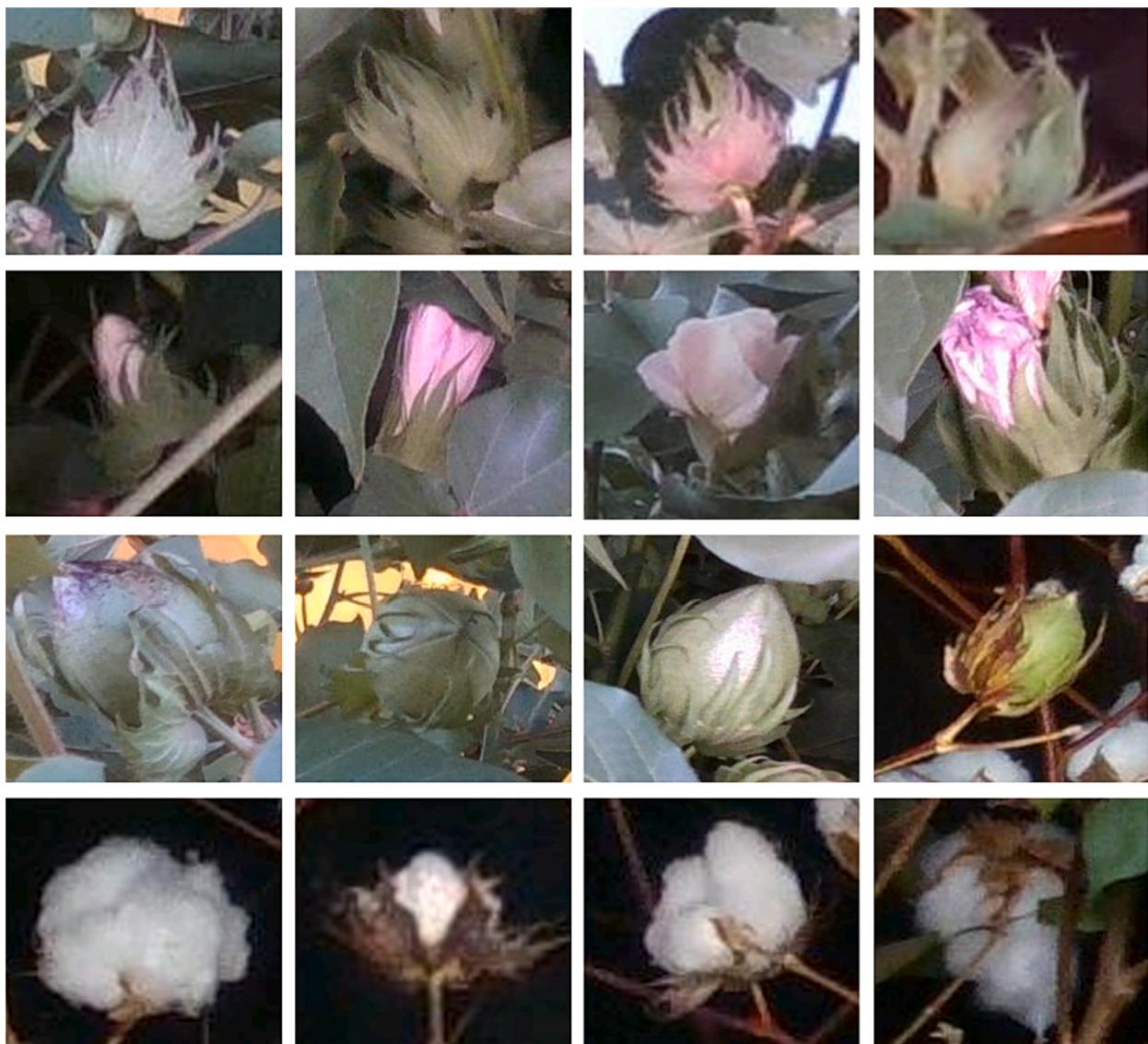
Instance segmentation techniques aim at pixel-level classification (semantic segmentation) of multiple instances of various objects in an image (Mueed Hafiz and Mohiuddin Bhat, 2020). In addition to detecting and enumerating cotton fruits in 2D, these techniques can also be used to localize the fruits in 3D point clouds of the rows, as explained in section 2.7. Consequently, state-of-the-art instance segmentation models such as Mask R-CNN (He et al., 2020) and YOLACT (Bolya et al., 2019) were considered for this study.

Mask R-CNN is a highly effective technique for instance segmentation, with a proven track record across a variety of applications (Mueed Hafiz and Mohiuddin Bhat, 2020). Mask R-CNN extends Faster R-CNN by adding a mask predictor head to its architecture. The model first employs the Faster R-CNN architecture to predict a class label and bounding box for an object based on regions proposed by the region proposal network (Ren et al., 2017). Then, a fully convolutional network (Shelhamer et al., 2017) generates a mask for the predicted bounding box, resulting in instance-level semantic segmentation. Despite not

being the fastest instance segmentation model, Mask R-CNN has shown to perform better on benchmark datasets than other state-of-the-art network architectures such as YOLACT (Bolya et al., 2019).

While the accuracy and speed of model inference were both of high importance for this study, the former was given more precedence for two reasons. First, the accuracy in enumerating the fruits clearly depended on the accuracy (not the speed) of segmentation and second, it was determined that the speed consideration could be satisfied in the network inference optimization phase of the project (further discussed in section 2.6). Therefore, Mask R-CNN was selected as the more suitable architecture for this study. Additionally, the proven performance of Mask R-CNN in segmenting OpenBolls (Adke et al., 2022) and Faster R-CNN in detecting Flowers (Jiang et al., 2020) further justify our choice of this model. It is worth noting that Mask R-CNN's ability to identify Squares and ClosedBolls or detect fruits in multiple growth-stages simultaneously had remained untested prior to this study.

To implement Mask R-CNN, the ResNet + FPN (Lin et al., 2017) architecture was used as the model's backbone network for feature extraction instead of the ResNet-C4 architecture proposed in the original Faster R-CNN implementation (Ren et al., 2017). ResNet + FPN has been shown to perform with higher accuracy in object detection and segmentation benchmarks (He et al., 2020; Lin et al., 2017). The lighter 50-layer ResNet was selected over the 101-layer ResNet due to the limited computational capabilities of the Xavier AGX on which the model was to be deployed. Additionally, to speed up training and leverage transfer learning from a larger dataset, the weights for the network were initialized with weights from a pre-trained model on the MS COCO dataset (Lin et al., 2014). Examples of the model's performance are



**Fig. 7.** Examples of images from each of the four categories. From top row to bottom, Square, Flower, ClosedBoll, and OpenBoll.

shown in Figs. 11 and 12.

#### 2.4. Model evaluation

A fruit's prediction by the model was considered a True Positive (TP) if it met two conditions. First, it belonged to the same category as the fruit's corresponding ground truth; and second, the prediction and ground truth's segmentations had an Intersection Over Union (IOU) of 0.5 or higher, which is the standard threshold. If either of these conditions was not met, the prediction was considered a False Positive (FP). Additionally, if no corresponding ground truth was found for the prediction, it was also considered a FP. A prediction was considered a False Negative (FN) if there was a ground truth in the image that did not have a corresponding prediction. Average Precision (AP) (Everingham et al., 2015), which is a standard evaluation metric for object detection and segmentation networks, was employed to assess the model's performance. AP is calculated as the area under the precision-recall curve, as shown in Eqs. (1)–(3).

$$\text{Precision } (P) = \frac{TP}{TP + FP} \quad (1)$$

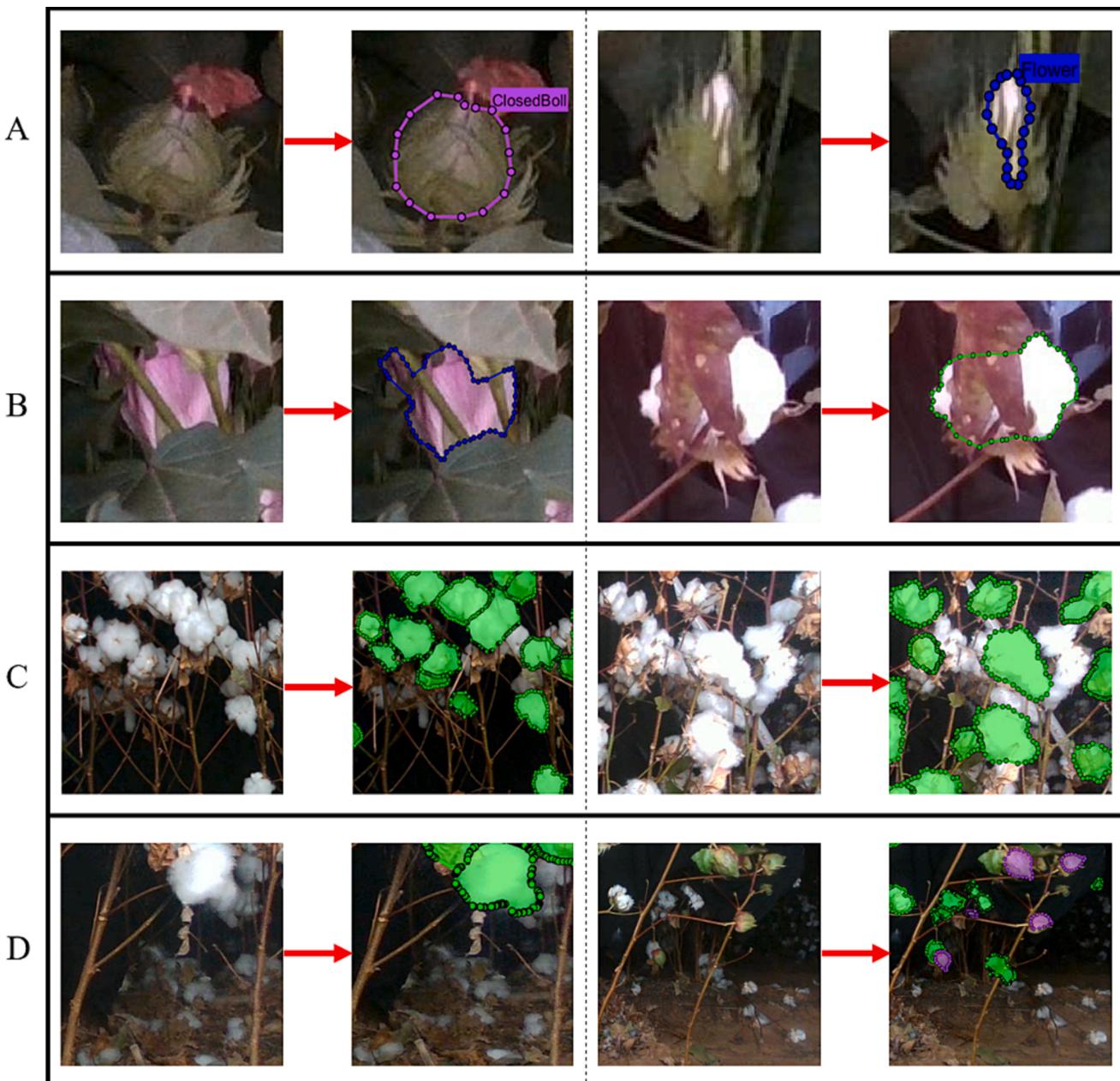
$$\text{Recall } (R) = \frac{TP}{TP + FN} \quad (2)$$

$$AP = \int_0^1 P(R) \, dR \quad (3)$$

#### 2.5. Fruit enumeration by tracking

A significant highlight of this work is its ability to track a segmented fruit throughout the image frames in which that fruit appears. This ability accomplishes two critical objectives. First, it avoids counting the same fruit multiple times. Second, it increases the classification accuracy by labeling the fruit in each frame, as its pose with respect to the camera changes. In the end, the fruit is assigned the label that was most frequently encountered in the ensemble of frames. Owing to its efficiency, ease of implementation, and robustness in the presence of occlusions, the commonly used tracking method, SORT (Bewley et al., 2016), was used as a baseline for the system's enumeration by tracking algorithm.

While the original implementation of the SORT algorithm is effective in enumerating objects within a single category based on the number of tracked objects in a video, it becomes impractical when input detections



**Fig. 8.** Illustration of established protocols for the annotation of images: A) Fruit transitional stages, examples include Flower to ClosedBoll and Square to Flower. – B) Obstructed Fruits – C) Overcrowded Scenes – D) Dropped fruits on the ground.

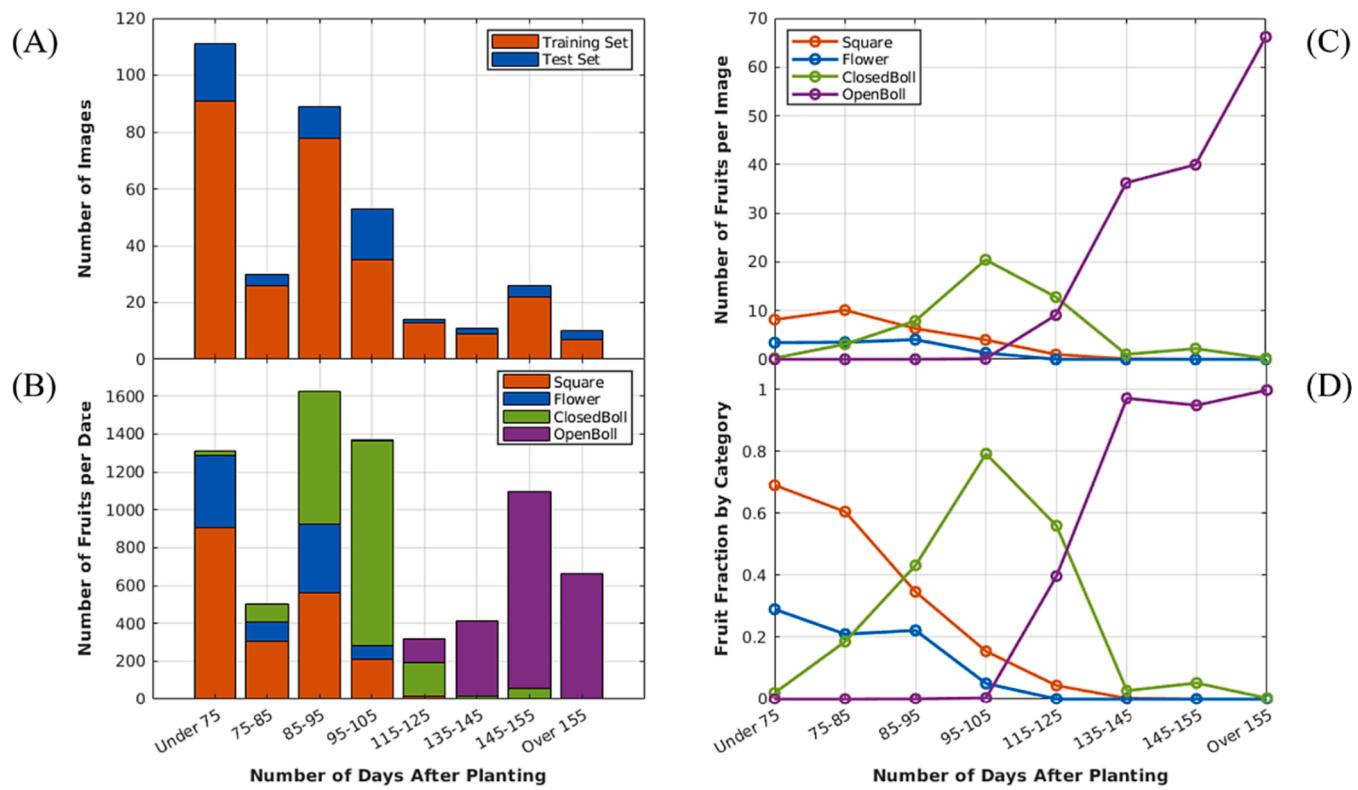
**Table 2**

A summary of constructed dataset.

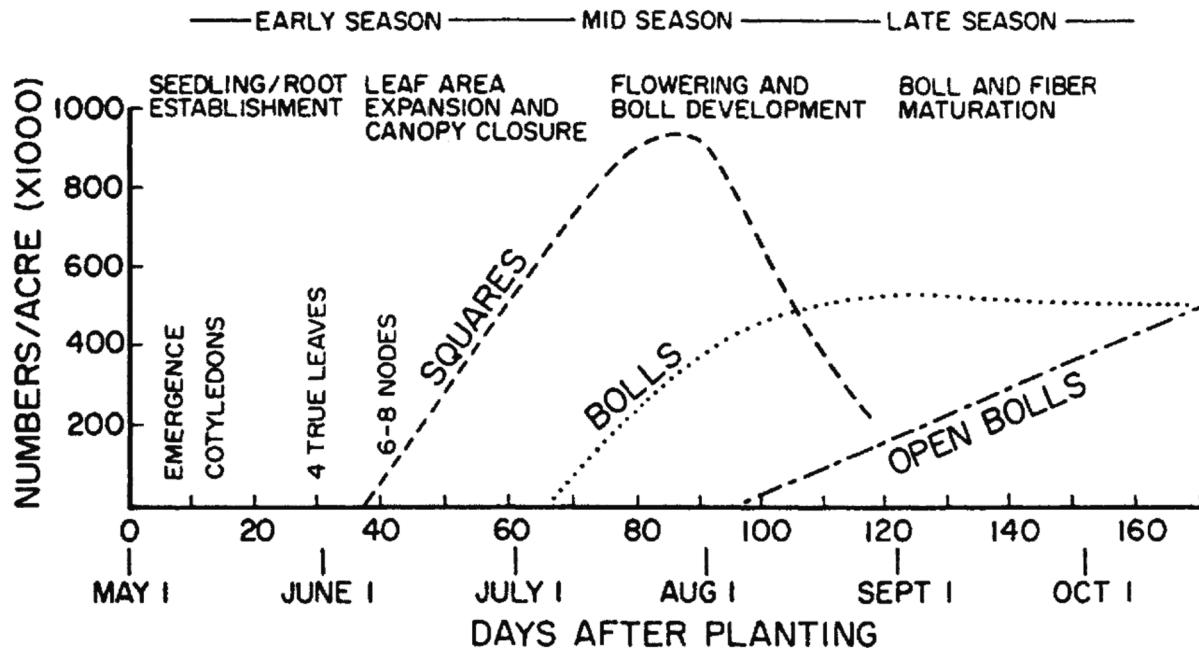
	Number of Images						Number of Instances			
	Dryland	30ET	90ET	Day	Night	Total	OpenBoll	ClosedBoll	Flower	Square
Training Set	186	49	46	134	147	281	1748	1508	636	1401
Test Set	51	5	7	23	40	63	484	644	278	595
Total	237	54	53	157	187	344	2232	2152	914	1996

belong to different categories. This is because SORT only takes into account the position of the detections within the frames and not their labels. Therefore, modifications were made to address this limitation, allowing it to effectively enumerate the four categories of fruits. The proposed algorithm is capable of online processing all the predicted fruits in parallel; however, for better clarity, the three stages of the algorithm are demonstrated below for a single fruit (Fig. 13):

**I. Initialization Stage:** When the fruit is detected by the model for the first time, a tracker that stores the fruit's predicted label and position (in the form of a bounding box) is initialized. The tracker will continue to preserve the predicted labels of the fruit in subsequent frames where it is detected. The tracker also includes an estimation model that predicts the position of the fruit in



**Fig. 9.** A) The distribution of the images in the training and test set with respect to days after planting (DAP). – B) The distribution of the instances (cotton fruits) in the dataset with respect to DAP. – C) The proportion of each category's count per image relative to DAP. – D) The proportion of each category's count relative to the total number of fruits.

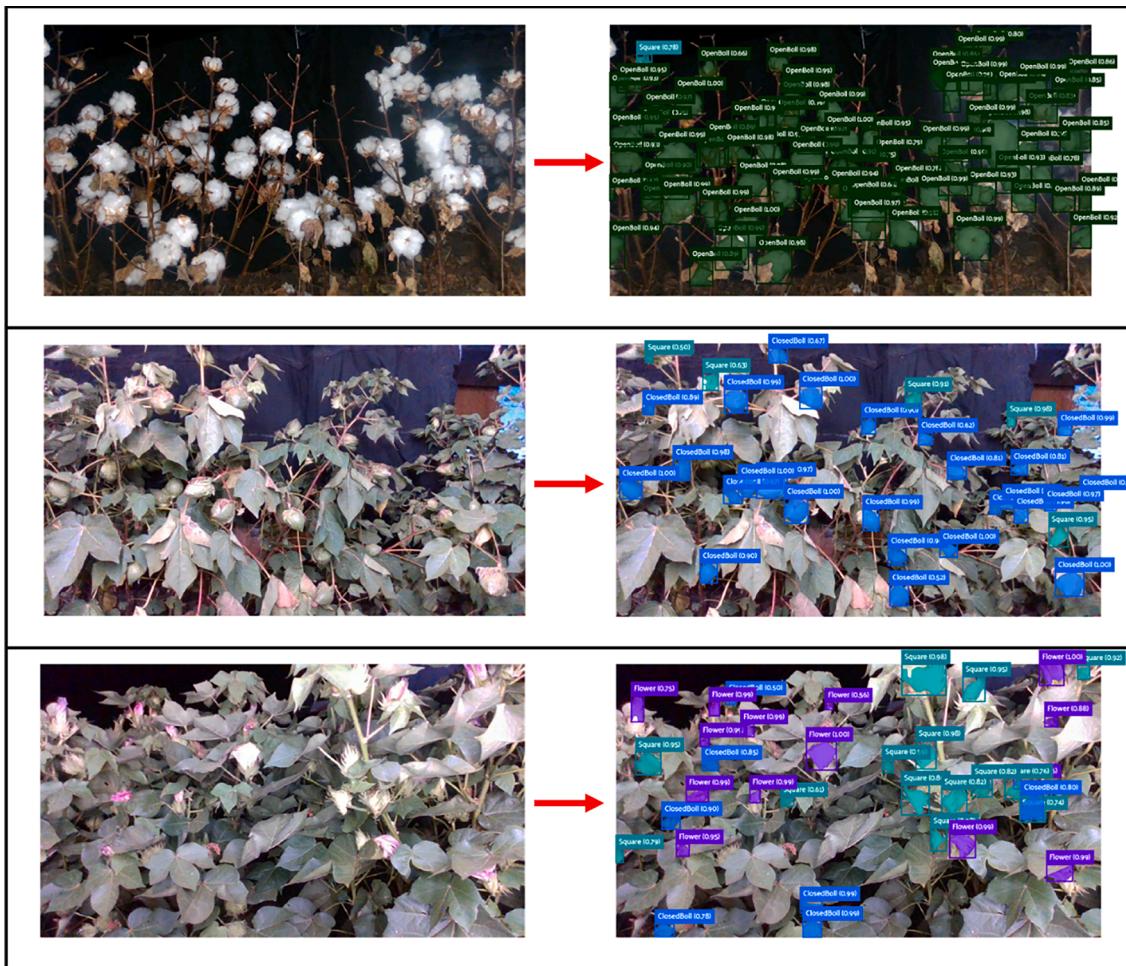


**Fig. 10.** Seasonal development of cotton in the Mid-South showing the production pattern of squares, bolls and open bolls (Oosterhuis. (1990), with permission from ACSESS).

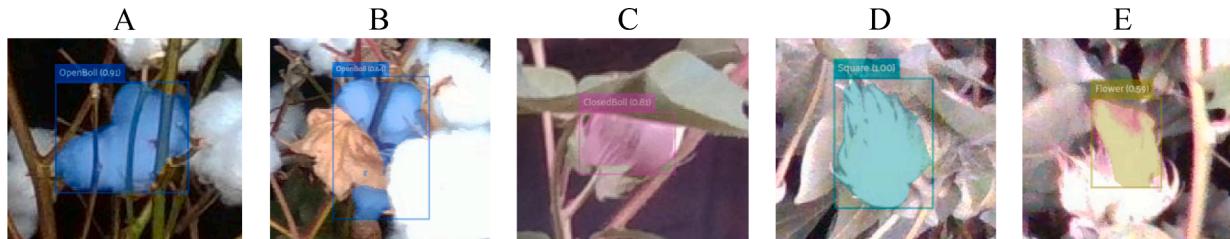
upcoming frames by extrapolating its motion using a Kalman Filter (Kalman, 1960), as originally proposed in SORT.

**II. Observation Stage:** In the following frames, an IOU matching algorithm (as implemented in SORT) attempts to match the prediction of the tracker with one of the model's predictions. If a

match is identified, the tracker's position is updated with the matched model prediction and then used to predict the fruit's position for the next frame. Meanwhile, the variables  $N_{cnt}$  and  $N_{obs}$  keep a record of the number of consecutive frames with and without a match, respectively. If  $N_{obs}$  reaches the user-defined



**Fig. 11.** Examples of the model's performance on crowded scenes from the test set.



**Fig. 12.** Examples of predictions for obstructed (A, B, and C) and intensity-saturated fruits (D and E).

threshold  $T_{obs}$ , it is assumed that the fruit can no longer be detected and the algorithm will proceed to its final stage.

**III. Enumeration Stage:** For a fruit to be counted, the highest value of  $N_{cnt}$ , which represents the maximum number of consecutive frames with a match ( $N_{enu}$ ), should meet a second user-defined threshold  $T_{enu}$ . If this condition is met, then the fruit category with the highest occurrence in the recorded labels will be counted (Fig. 14B). Finally, the tracker will be terminated for memory management.

The success of the proposed algorithm relies on setting appropriate values for  $T_{enu}$  and  $T_{obs}$ , which are determined by the speed at which the camera is moving and its frame rate. The moving speed will determine the number of frames,  $N$ , in which each fruit can appear. To optimize the algorithm's performance, empirical analysis of videos captured at different moving speeds suggested that the values for  $T_{enu}$  and  $T_{obs}$

should be roughly set to  $N/3$  and  $N$ , respectively. In the data collection phase (section 2.1), it was observed that each fruit was visible to the camera for approximately 10 s at a tractor speed of 0.1—0.15 m/s. Therefore, based on a frame rate of 30 FPS,  $T_{enu}$  and  $T_{obs}$ , were set to 100 and 300 frames, respectively.

$T_{enu}$  is particularly essential in avoiding the premature enumeration of fruits in the event of FP predictions. Non-fruit objects, such as leaves, are rarely predicted for long sequences of consecutive frames. Therefore, they would not be counted as a fruit since  $T_{enu}$  is not met in these scenarios (Fig. 14A). Similarly,  $T_{obs}$  is crucial in preventing the premature termination of trackers. For instance, when a fruit that is already being tracked gets temporarily obstructed from camera's field of view and the model fails to detect it, the tracker will continue to predict the fruit's position in the next frames until the fruit is detected again (Fig. 14C).

To accurately evaluate the enumeration by tracking algorithm, it is necessary to manually count the cotton fruits in the rows where data was

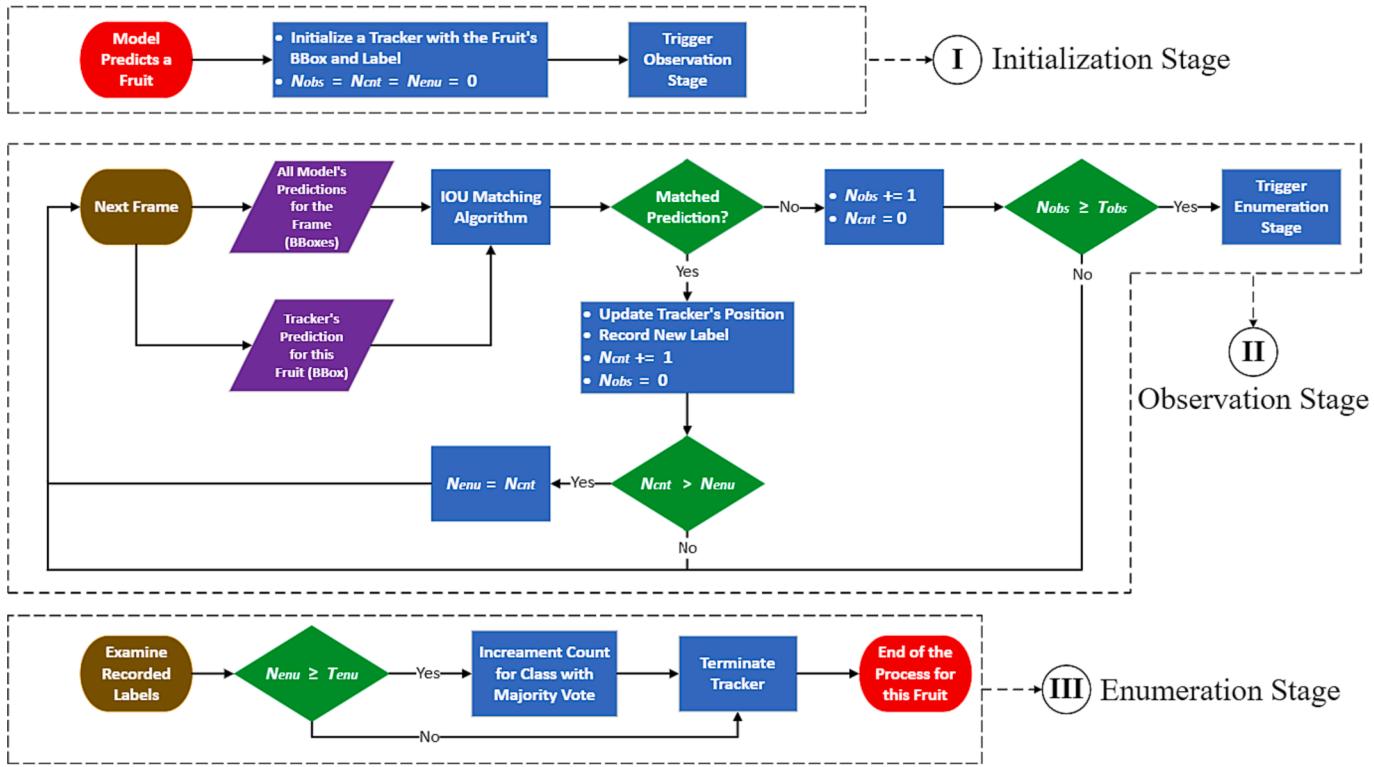


Fig. 13. A flowchart of the enumeration by tracking algorithm for a detected fruit.

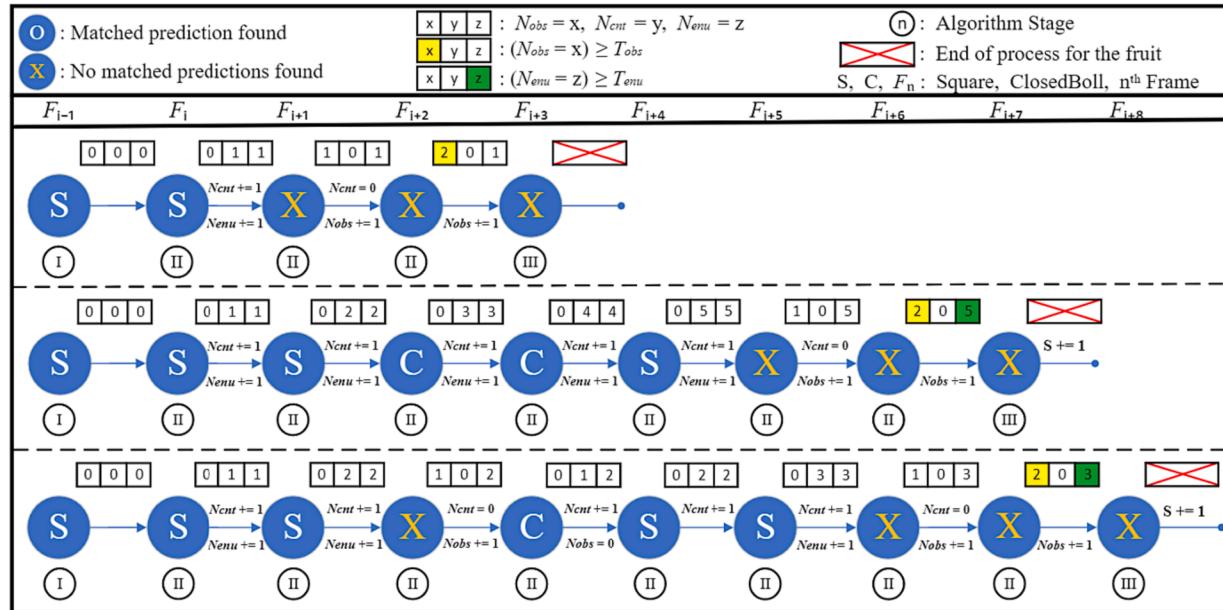


Fig. 14. Example scenarios of the algorithm's behavior when facing a A) FP prediction that appear and disappears quickly – B) a fruit that is consecutively detected but with different category classifications – C) a fruit that was incorrectly not detected (FN) at  $F_{i+2}$ .  $T_{enu}$  and  $T_{obs}$  are set to 2 and 3, respectively.

collected. However, due to time constraints, crowded plant conditions, and copious number of fruits in the experimental field (section 2.1), it was not feasible to hand-count them for evaluation purposes. Therefore, additional data was collected from September 9th to October 12th of 2022. A total of 16 videos were collected from rows in dryland conditions with lower populations of leaves and fruits, allowing for a precise manual counting; see Fig. 15.

## 2.6. Network optimization in TensorRT

The computing power of a compact edge device, such as AGX Xavier, is significantly more limited than that of a workstation GPU typically used for neural network training. As a result, the complexity of the model that can be deployed on these devices is limited, and their latency is much higher. However, Xavier GPU architectures are equipped with computing cores that support half-precision floating-point (FP16) and 8-bit integer (INT8) formats, which can be leveraged to optimize the



**Fig. 15.** Snapshots of videos taken for evaluation of the enumeration by tracking algorithm. Videos were taken both in daylight (the two images on the left and middle) and at night with artificial lighting (the image on the right).

network in TensorRT for faster inference speed and reduced memory usage. TensorRT is a deep learning inference optimizer and runtime library, developed by NVIDIA, that is designed to accelerate the performance of neural networks on NVIDIA GPUs. It provides a high-performance inference engine that optimizes and deploys trained neural networks for use in production environments. A case in point is the work by Yanchao Zhang et al. (2022) showcasing the use of TensorRT to optimize their novel RTSD-Net model for real-time strawberry detection on a NVIDIA Jetson device. The application of TensorRT led to a twofold acceleration in model inference speed while maintaining accuracy levels within a negligible 1 % difference.

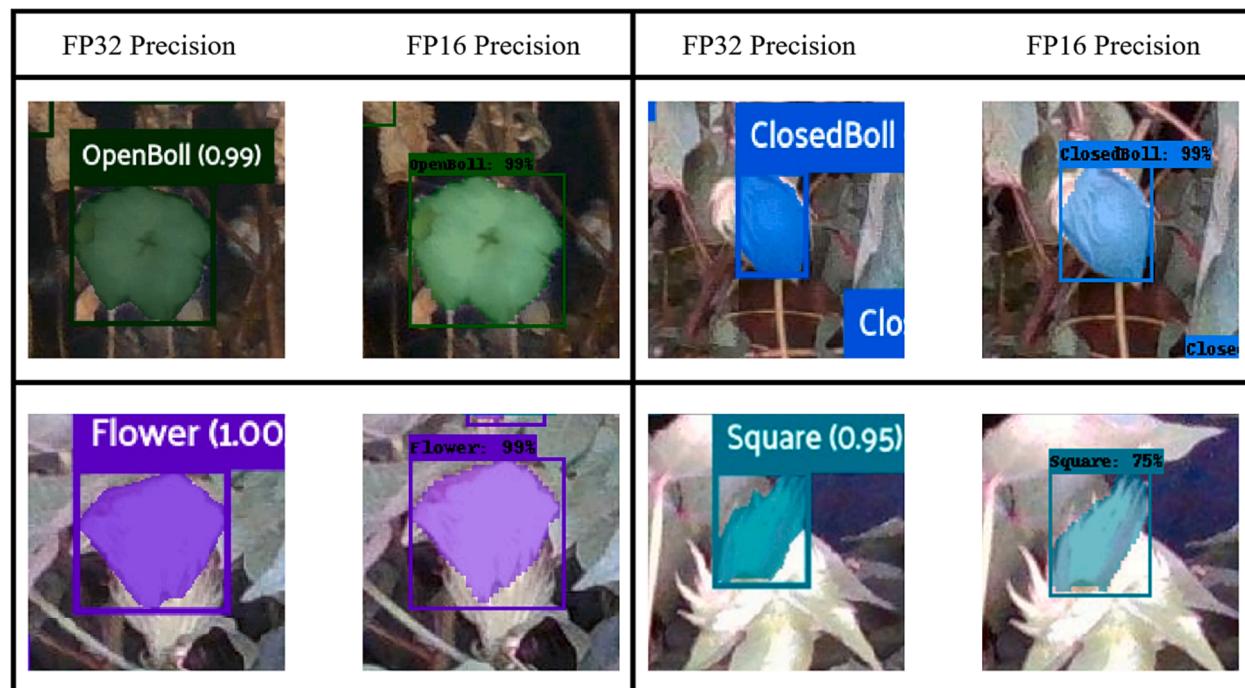
Quantizing the model weights from single-precision floating-point (FP32) to lower precisions, such as FP16 or INT8, can significantly accelerate inference. It has been shown that networks optimized with FP16 precision can achieve approximately the same accuracies to those with FP32 precision, while INT8 precision can result in slightly lower accuracies due to its more limited range of precision (Gysel et al., 2016). Therefore, to prioritize accuracy as discussed in section 2.3, the developed Mask-RCNN model was optimized with FP16 precision.

The TensorRT model with FP16 precision was evaluated for fruit segmentation using the same dataset and metrics demonstrated in section 2.4. The results showed that the new model achieved an identical

AP score while offering a significant improvement in inference speed. Specifically, the TensorRT model was approximately 2.4 times faster in inference than the original Mask R-CNN model (Fig. 16).

### 2.7. Three-Dimensional segmentation by projection

Pixel-level classification of the RGB images obtained from the developed instance segmentation model can be further utilized to segment fruits in the generated 3D point clouds (section 2.1). By leveraging the depth information associated with each pixel in the RGB-D images, the segmented regions can be mapped onto their corresponding 3D points. To achieve this, it is necessary to obtain D435i's camera matrix,  $K$ , which was stored from RTAB-Map during the data collection phase. As shown in Eq. (4),  $K$  provides information about the internal parameters of the camera, including the focal length ( $f_x$  and  $f_y$ ) and principal point ( $c_x$  and  $c_y$ ). The focal length represents the distance between the camera's pinhole and the image plane, determining object magnification in the image, while the principal point indicates the location where the optical axis of the camera intersects with the image plane.



**Fig. 16.** A comparison between the segmentation results obtained by the original model with FP32 precision and the new model with FP16. It can be observed that the generated masks from the FP32 model often appear marginally more refined than those of the FP16 model.

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

To map a classified pixel,  $p(u, v)$ , in the RGB-D image onto its corresponding 3D point,  $p(x, y, z)$ , Eqs. (5)–(8) can be used, where  $d(u, v)$  denotes value of the  $p(u, v)$  in the depth image (Hartley and Zisserman, 2004).

$$z = d(u, v) \quad (5)$$

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \frac{1}{z} \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (6)$$

$$\Rightarrow \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{x \cdot f_x + z \cdot c_x}{z} \\ \frac{y \cdot f_y + z \cdot c_y}{z} \\ 1 \end{bmatrix} \quad (7)$$

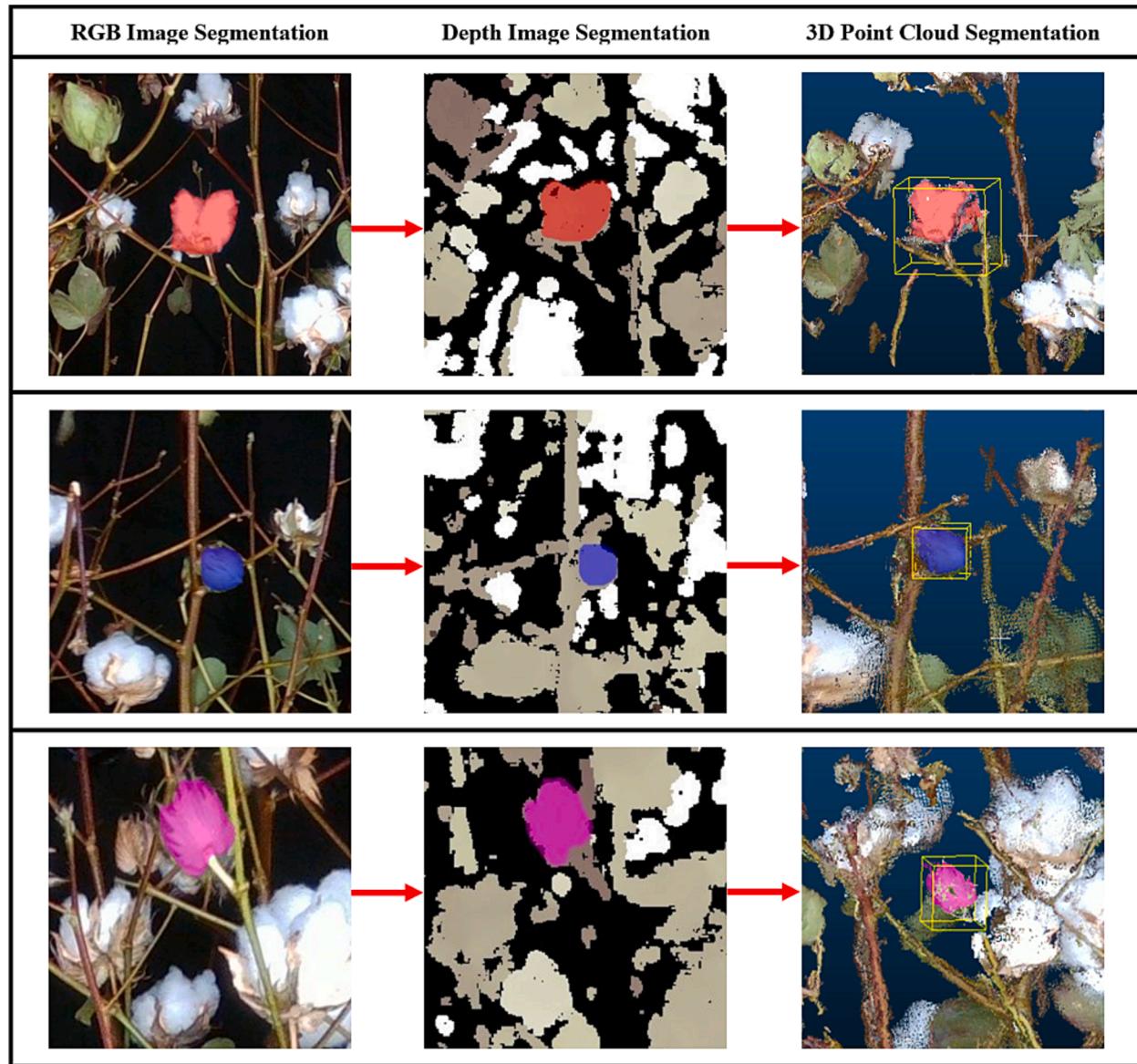
$$\Rightarrow \begin{cases} x = \frac{(u - c_x) \cdot z}{f_x} \\ y = \frac{(v - c_y) \cdot z}{f_y} \end{cases} \xrightarrow{z = d(u, v)} \begin{cases} x = \frac{(u - c_x) \cdot d(u, v)}{f_x} \\ y = \frac{(v - c_y) \cdot d(u, v)}{f_y} \end{cases} \quad (8)$$

To illustrate this process, Figs. 17 and 18 provide examples of segmentations being projected from 2D to 3D.

### 3. Results and discussion

#### 3.1. Instance segmentation model

The model achieves an AP score of 79 % for both segmentation and detection tasks (detailed in Table 3), which is on par with the leading scores reported on the MS COCO dataset. In comparison to other deep learning models mentioned in Table 1, the model outperforms the Flower detection model proposed by Jiang et al. (2020) by 5 %. Sun et al. (2022) and Yan Zhang et al. (2022) report higher AP scores for ClosedBoll detection; however, it should be emphasized that their scores are



**Fig. 17.** Process of segmentations being projected from 2D and 3D by leveraging depth information from RGB-D image. The rows demonstrate this process for an OpenBoll, ClosedBoll, and a Square from top to bottom.



**Fig. 18.** Projected segmentations of the developed model from 2D to the 3D point cloud shown in Fig. 5B.

**Table 3**

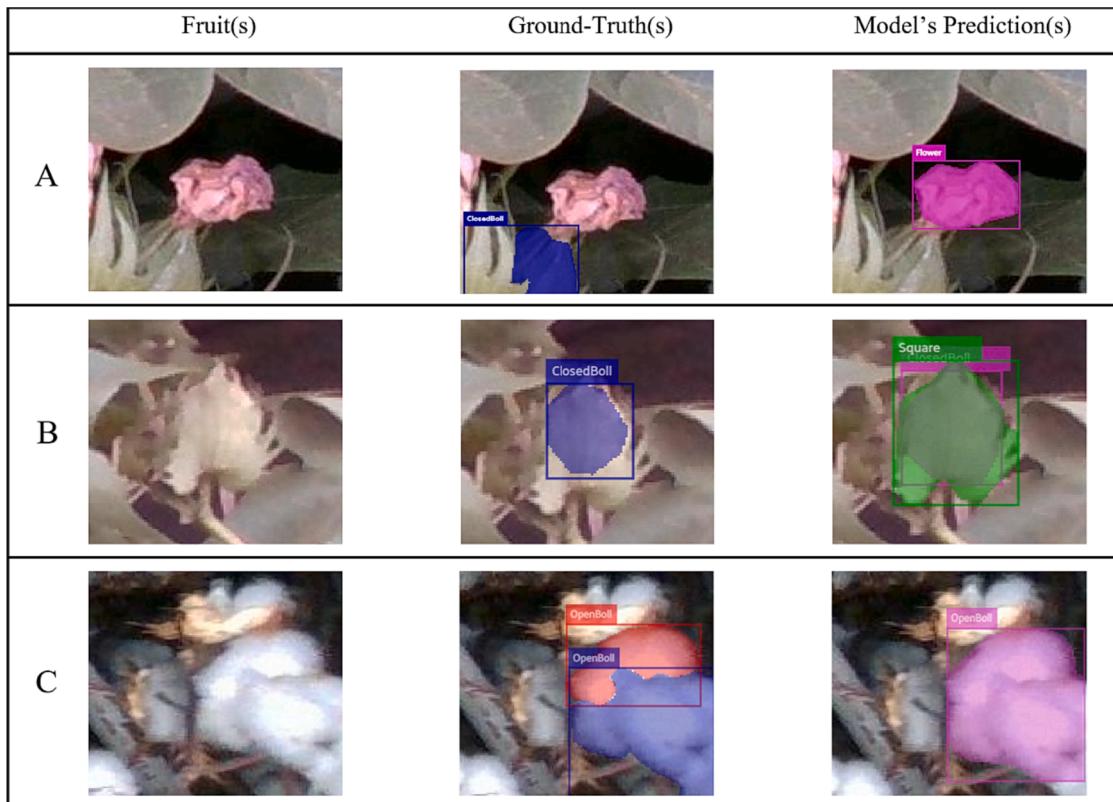
AP scores across fruit categories in both detection and segmentation.

	2D Detection					2D Segmentation				
	OpenBoll	ClosedBoll	Flower	Square	Average	OpenBoll	ClosedBoll	Flower	Square	Average
AP	85.4 %	79.5 %	76.9 %	74.0 %	78.9 %	82.9 %	78.7 %	78.2 %	75.0 %	78.7 %

not directly comparable to those in this study. Notably, Sun et al. (2022) used single points in their custom model to localize the ClosedBolls instead of bounding boxes, resulting in alternative definitions for TP, FP, FN, and thus, the AP score. Additionally, the authors developed a Faster-RCNN model for comparison and reported an AP score of 38.5 %, which is significantly lower than the detection score of ClosedBolls reported in Table 3. Similarly, Yan Zhang et al. (2022) used a nonstandard definition of AP by dividing the sum of precision values for each image

by the total number of images. As for Adke et al. (2022), the authors did not report an evaluation based on the AP score. Instead, they quantified the performance of their detection model based solely on its enumeration results.

Regarding the performance of the developed model in this study, two additional points should be emphasized. First, it is important to reiterate that no other studies have undertaken detection and segmentation of cotton fruits across multiple stages of growth. Second, the model's



**Fig. 19.** A) FP classification for the Flower prediction. – B) FP classification for Square prediction. – C) One FP and two FN classifications for the OpenBoll prediction.

results could be objectively improved were it not for the uncertainty surrounding the validity of false predictions in the following scenarios:

- I. Occasionally, flowers can remain attached to a developing boll after the growth cycle which is referred to as a *bloom tag* (Ritchie et al., 2007). Depending on the scene, domain experts may have varying interpretations of whether the Flower has fully transitioned to a ClosedBoll, which could result in a FP classification; see Fig. 19A.
- II. Developing bolls that are obscured by their bracts may result in both Square and ClosedBoll predictions. In such cases, it may be difficult to determine whether the target fruit is in the ClosedBoll or Square stage, resulting in one of the predictions being classified as a FP; see Fig. 19B.
- III. As mentioned in the previous section, there exist crowded scenes in which only fruits with identifiable boundaries were annotated. Due to the complexity of these scenes, model predictions may not align with the given ground truth, but it could still be argued that the predictions correctly detected and classified the target fruits while complying with the established annotation protocol; see Fig. 19C.

Therefore, to further evaluate the model's predictions, a survey was conducted among 10 experts in the cotton physiology field, which included 80 unannotated images sampled temporally from the collected dataset in section 2.1 (these images were not used in the model development process). Each expert was asked to count the number of fruits for each category. To highlight the disparity in scene interpretation among the experts referred to in (I), the normalized values of the participants' counts were compared using the Relative Difference equation (Törnqvist et al., 1985), as shown in Eq. (9).

$$\text{Relative Difference } (\%) = \left| \frac{x - y}{\max(|x|, |y|)} \right| \bullet 100 \quad (9)$$

In Table 4, it can be observed that in the same image, the average relative difference among participants' counts is 27 % with the Square class exhibiting the largest discrepancy.

The experts' counts for each image were then compared to the model's counts using the Coefficient of Determination ( $R^2$ ) as an evaluation metric. The results, depicted in Fig. 20, show that the model achieves a notable  $R^2$  value of 0.97 at counting the total number of fruits in each image and an average of 0.85 for each category. Fig. 20 also demonstrates that the model's predictions align more closely with the experts' counts for the OpenBolls and ClosedBolls categories, while exhibiting larger discrepancies for the Flower and Square categories.

### 3.2. Fruit enumeration by tracking

The enumeration algorithm's performance was evaluated using the  $R^2$  metric, comparing the results to the hand-counted ground truth; see Table 5 and Fig. 21. The algorithm achieves a strong average  $R^2$  value of 0.93. For comparison, while the highest reported  $R^2$  measurement for OpenBoll enumeration by Sun et al. (2020) is 0.91, the proposed algorithm achieves an  $R^2$  value of 0.97. Furthermore, these findings highlight that, in addition to enhancing the classification accuracy of the segmentation model (as illustrated in section 2.5), the enumeration by tracking algorithm also improves the model's detection accuracy. This synergy is evident in the results, where the combined use of this algorithm and the model yielded  $R^2$  values of 0.95, 0.93, and 0.88 for the

**Table 4**

This table shows the average Relative Difference in fruit count for each image among the participants in the survey.

	OpenBoll	ClosedBoll	Flower	Square	Average
Relative Difference	8 %	28 %	28 %	43 %	27 %

detection and counting of ClosedBolls, Flowers, and Squares over videos, respectively. In contrast, when relying solely on the model, the per-image count for these three categories yielded lower  $R^2$  values of 0.93, 0.79, and 0.73 (Fig. 20).

The developed algorithm, while proficient in accurately counting the detected fruits by the instance segmentation model, exhibits limitations when applied in highly irrigated cotton fields characterized by dense, leaf-crowded canopies. The algorithm's performance is hindered by the challenges posed by obscured fruits hidden behind leaves. Since the segmentation model cannot detect fruits concealed in such environments, the enumeration algorithm faces difficulties in tracking and counting these obscured fruits effectively. Consequently, the algorithm's accuracy may be compromised in scenarios where fruits are extensively concealed, impacting the overall precision of the fruit counting process.

### 3.3. Three-Dimensional segmentation by projection

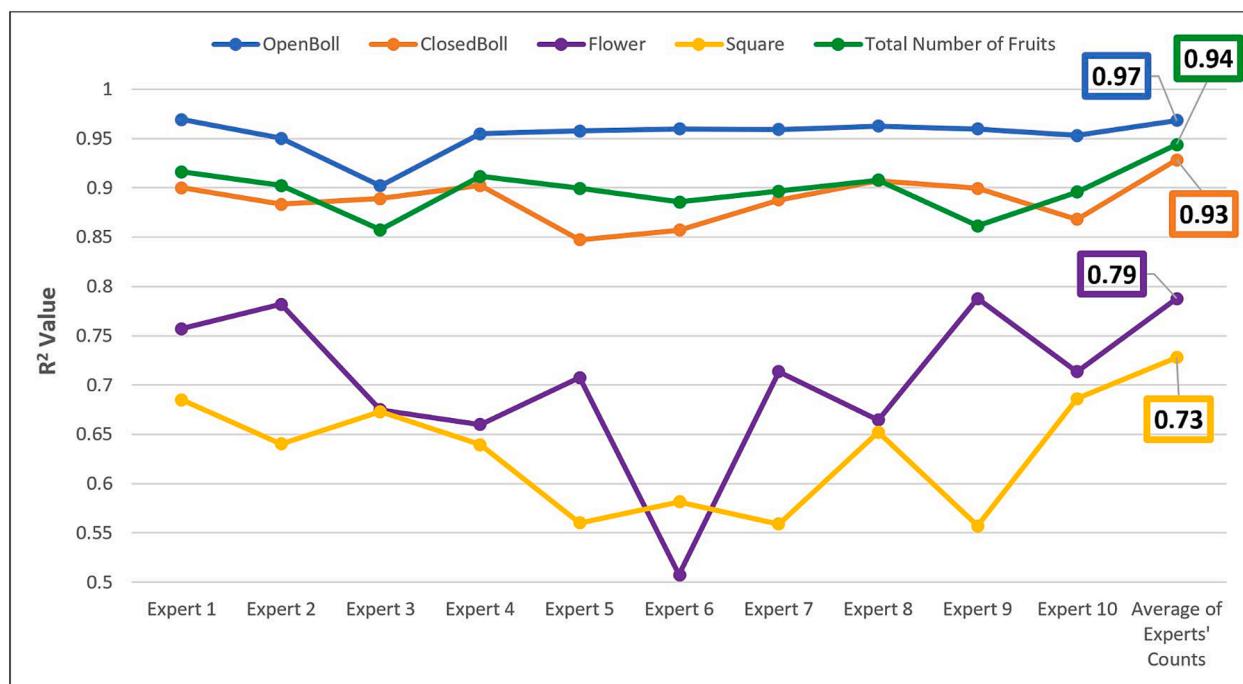
Since a shroud was only added in the later versions of the data acquisition system (section 2.1), the dataset (section 2.2) encompassed images captured in two distinct lighting conditions: one where the shroud minimizes color saturation from direct sunlight and another without any shading. While the 2D segmentation model was trained using images from both scenarios and demonstrated its capability to detect fruits under both conditions, the accuracy of the 3D segmentation heavily depends on the presence of the shroud. The point cloud generation system yields the most precise results when the shroud is utilized, highlighting a limitation wherein the absence of shade during data collection compromises the accuracy of the 3D segmentation output. Despite the robustness of the 2D segmentation model, this reliance on external conditions underscores a constraint in the system, emphasizing the need for consistent shading to ensure optimal 3D segmentation accuracy.

## 4. Conclusions

This paper presented CottonSense, a cost-effective and power-efficient HTP system for cotton plants that operates entirely on an NVIDIA Jetson AGX Xavier, for both data collection and processing. CottonSense offers an:

- I. Effective framework for collecting phenotypic measurements under agronomic field conditions using RGB-D cameras in the daylight and at night. During data collection, the system produces a 3D point cloud of each row in real-time by using a custom configuration for the RTAB-Map library.
- II. Approach for 2D and 3D segmentation of cotton fruits using a Mask-RCNN model at four different growth stages including square, flower, closed boll, and open boll. The model is optimized with TensorRT to process images 2.4 times faster during deployment, and it yields a strong agreement with the per image counts of ten field experts, as indicated by an average  $R^2$  value of 0.94.
- III. Efficient and accurate algorithm for enumeration of segmented fruits over large population of plants across consecutive video frames. The algorithm is developed upon SORT's tracking strategy and achieves an  $R^2$  value of 0.93 when compared to hand-counted fruits in the field.

The ability to enumerate and discriminate cotton fruit could have significant value for both breeders and producers. Cotton breeding is designed around not only yield, but also acclimation to environment, which is described by fruiting pattern. CottonSense enables improved capture of crop development and ontogeny, providing breeders with valuable insights into the growth patterns and stages of cotton crops. By harnessing the capabilities of CottonSense, breeders can develop robust yield models based on precise fruit timing and development, thereby



**Fig. 20.** The figure depicts the  $R^2$  measurement between the model's and expert's count for each category across all the images.

**Table 5**  
Evaluation of the enumeration by tracking algorithm based on  $R^2$ .

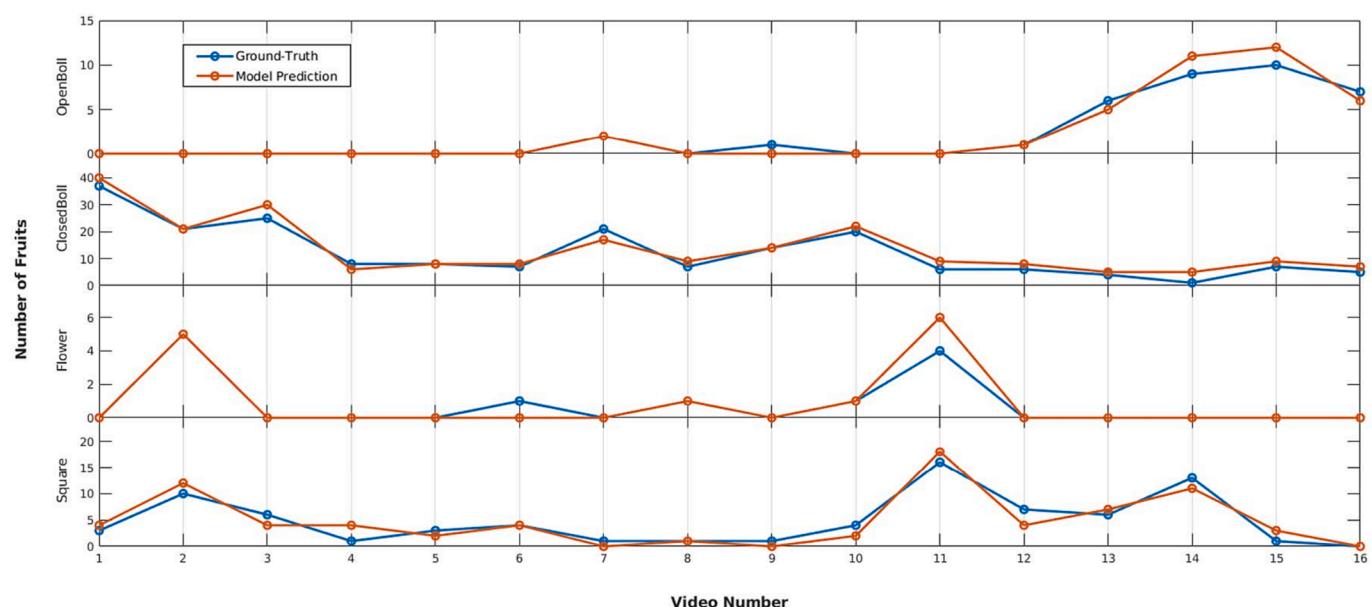
	OpenBoll	ClosedBoll	Flower	Square	Average
$R^2$	0.97	0.95	0.93	0.88	0.93

optimizing crop productivity. To further enhance the accuracy of crop assessment and evaluation, future efforts will be directed towards expanding the system's capabilities through the integration of additional phenotypic parameters, including leaf counting and node counting. Ongoing research and development of CottonSense aim to refine its functionalities to better serve breeders and producers, ultimately maximizing cotton yields. This includes the development of more

resilient cotton varieties capable of withstanding challenging conditions such as water limitation.

#### Author Contributions

FB, HS, and GR conceived the original idea; FB designed and performed the experiments, and wrote the manuscript; YK contributed to software development and conducting the experiments; IP contributed to data collection and analysis; GR and HS provided critical feedback that shaped the research and the manuscript.



**Fig. 21.** Comparison of the counts by the enumeration algorithm and the hand-counted ground-truth in the collected 16 videos.

## CRediT authorship contribution statement

**Farshad Bolouri:** Conceptualization, Methodology, Validation, Data curation, Writing – original draft, Visualization. **Yildirim Kocoglu:** Software. **Irish Lorraine B Pabuayon:** Formal analysis, Investigation. **Glen Lorin Ritchie:** Conceptualization, Writing – review & editing, Supervision. **Hamed Sari-Sarraf:** Conceptualization, Writing – review & editing, Supervision.

## Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: [Farshad Bolouri reports financial support and equipment, drugs, or supplies were provided by BASF.]

## Data availability

The data that has been used is confidential.

## Acknowledgement

This work was funded by BASF. The authors would like to thank the University of Georgia cotton team, under the direction of Dr. John L Snider, for participating in the model evaluation survey (section 3.1). The authors would also extend their appreciation to Ethan Elliott, Long Duong, and Alyson Willis for their contribution to the data annotation process.

## Appendix A. Supplementary material

Supplementary data to this article can be found online at <https://doi.org/10.1016/j.compag.2023.108531>.

## References

- Adke, S., Li, C., Rasheed, K.M., Maier, F.W., 2022. Supervised and Weakly Supervised Deep Learning for Segmentation and Counting of Cotton Bolls Using Proximal Imagery. Sensors 22. <https://doi.org/10.3390/s22103688>.
- Besl, P.J., McKay, N.D., 1992. A Method for Registration of 3-D Shapes. IEEE Trans. Pattern Anal. Mach. Intell. 14, 239–256. <https://doi.org/10.1109/34.121791>.
- Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B., 2016. Simple online and realtime tracking. In: Proc. - Int. Conf. Image Process. ICIP 2016-Augus, pp. 3464–3468. <https://doi.org/10.1109/ICIP.2016.7533003>.
- Bolya, D., Zhou, C., Xiao, F., Lee, Y.J., 2019. YOLACT Better Real-Time Instance Segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 44, 1108–1121. <https://doi.org/10.1109/ICCV.2019.00925>.
- Constable, G.A., Bangs, M.P., 2015. The yield potential of cotton (*Gossypium hirsutum* L.). F. Crop. Res. 182, 98–106. <https://doi.org/10.1016/j.fcr.2015.07.017>.
- Everingham, M., Eslami, S.M.A., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A., 2015. The Pascal Visual Object Classes Challenge: A Retrospective. Int. J. Comput. vis. 111, 98–136. <https://doi.org/10.1007/s11263-014-0733-5>.
- Fangueiro, R., Rana, S., 2016. Natural Fibres: Towards Industrial and Technology Advances in Science Applications. RILEM Bookseries. [https://doi.org/10.1007/978-94-017-7515-1\\_31](https://doi.org/10.1007/978-94-017-7515-1_31).
- Feng, A., Zhang, M., Sudduth, K.A., Vories, E.D., Zhou, J., 2019. Cotton yield estimation from UAV-based plant height. Trans. ASABE 62, 393–404. <https://doi.org/10.13031/trans.13067>.
- Gysel, P., Motamedi, M., Ghiasi, S., 2016. Hardware-oriented approximation of convolutional neural networks, pp. 1–8. <https://doi.org/10.48550/arXiv.1605.06402>.
- Hartley, R., Zisserman, A., 2004. Multiple View Geometry in Computer Vision, 2nd ed. Cambridge University Press. <https://doi.org/10.1017/CBO9780511811685.014>.
- He, K., Gkioxari, G., Dollár, P., Girshick, R., 2020. Mask R-CNN. IEEE Trans. Pattern Anal. Mach. Intell. 42, 386–397. <https://doi.org/10.1109/TPAMI.2018.2844175>.
- Huang, Y., Brand, H.J., Sui, R., Thomson, S.J., Furukawa, T., Ebelhar, M.W., 2016. Cotton yield estimation using very high-resolution digital images acquired with a low-cost small unmanned aerial vehicle. Trans. ASABE 59, 1563–1574. <https://doi.org/10.13031/trans.59.11831>.
- Intel, 2023. Intel® RealSense™ Product Family D400 Series. Available at: <https://dev.intelrealsense.com/docs/intel-realsense-d400-series-product-family-datasheet>.
- Jiang, Y., Li, C., Xu, R., Sun, S., Robertson, J.S., Paterson, A.H., 2020. DeepFlower: a deep learning-based approach to characterize flowering patterns of cotton plants in the field. Plant Methods 16, 1–17. <https://doi.org/10.1186/s13007-020-00698-y>.
- Kalman, R.E., 1960. A new approach to linear filtering and prediction problems. J. Fluids Eng. Trans. ASME 82, 35–45. <https://doi.org/10.1115/1.3662552>.
- Labbé, M., Michaud, F., 2019. RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation. J. f. Robot. 36, 416–446. <https://doi.org/10.1002/rob.21831>.
- Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S., 2017. Feature Pyramid Networks for Object Detection. CVPR. <https://doi.org/10.1109/ICCVS.2019.00110>.
- Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C. L., 2014. Microsoft COCO: Common objects in context. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics) 8693 LNCS, 740–755. [https://doi.org/10.1007/978-3-319-10602-1\\_48](https://doi.org/10.1007/978-3-319-10602-1_48).
- MathWorks, 2018. Image Labeler. Available at: <https://www.mathworks.com/help/vision/ref/imagelabeler-app.html>.
- Mued Hafiz, A., Mohiuddin Bhat, G., 2020. A Survey on Instance Segmentation. Int. J. Multimed. Inf. Retr. 9, 171–189. <https://doi.org/10.1007/s13735-020-00195-x>.
- Muja, M., Lowe, D.G., 2009. Fast approximate nearest neighbors with automatic algorithm configuration. VISAPP 2009 - Proc. 4th Int. Conf. Comput. vis. Theory Appl. 1, 331–340. <https://doi.org/10.5220/0001787803310340>.
- Normanly, J., 2012. High-Throughput Phenotyping in Plants. Methods and Protocols. [https://doi.org/10.1007/978-1-61779-995-2\\_11](https://doi.org/10.1007/978-1-61779-995-2_11).
- NVIDIA, 2016. Efficient inference with tensorrt 1–24. Available at: <https://developer.nvidia.com/tensorrt>.
- NVIDIA, 2019. NVIDIA Jetson AGX Xavier. Available at: <https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-agx-xavier/>.
- Oberti, R., Shapiro, A., 2016. Advances in robotic agriculture for crops. Biosyst. Eng. 146, 1–2. <https://doi.org/10.1016/j.biosystemseng.2016.05.010>.
- Oosterhuis, D.M., 1990. Growth and Development of a Cotton Plant. Nitrogen Nutr. Cott. Pract. Issue 1–24. <https://doi.org/10.2134/1990.nitrogennutritionofcotton.c1>.
- Pabuayon, I.L.B., Sun, Y., Guo, W., Ritchie, G.L., 2019. High-throughput phenotyping in cotton: a review. J. Cott. Res. 2, 1–9. <https://doi.org/10.1186/s42397-019-0035-0>.
- Pabuayon, I.L.B., Kelly, B.R., Mitchell-McCallister, D., Coldren, C.L., Ritchie, G.L., 2021. Cotton boll distribution: A review. Agron. J. 113, 956–970. <https://doi.org/10.1002/agj2.20516>.
- Park, J., Zhou, Q.-Y., Koltun, V., 2017. Colored Point Cloud Registration Revisited. Iccv. <https://doi.org/10.1109/ICCV.2017.25>.
- 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. Precis. Agric. 18, 224–244. <https://doi.org/10.1007/s11119-016-9458-5>.
- Rahmemoonfar, M., Sheppard, C., 2017. Deep count: Fruit counting based on deep simulated learning. Sensors (switzerland) 17, 1–12. <https://doi.org/10.3390/s17040905>.
- Ren, S., He, K., Girshick, R., Sun, J., 2017. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans. Pattern Anal. Mach. Intell. 39, 1137–1149. <https://doi.org/10.1109/TPAMI.2016.2577031>.
- Ritchie, G.L., Bednarz, C.W., Jost, P.H., Brown, S.M., 2007. Cotton Growth and Development. <https://hdl.handle.net/10724/12192>.
- Ritchie, G.L., Whitaker, J.R., Collins, G.D., 2012. Effect of sample size on cotton plant mapping analysis and results. J. Cotton Sci. 16, 224–232. Available at: <https://www.cotton.org/journal/2011-15/3/upload/Agronomy-002-Ritchie-ERRATA-11-12-DRAFT1.pdf>.
- Rublee, E., Rabaud, V., Konolige, K., Bradski, G., 2011. ORB: An efficient alternative to SIFT or SURF. Proc. IEEE Int. Conf. Comput. vis. 2564–2571 <https://doi.org/10.1109/ICCV.2011.6126544>.
- Shelhamer, E., Long, J., Darrell, T., 2017. Fully Convolutional Networks for Semantic Segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 39, 640–651. <https://doi.org/10.1109/TPAMI.2016.2572683>.
- Shi, J., Tomasi, C., 1994. Good Features to Track. Image, Rochester, N.Y, pp. 593–600. <https://doi.org/10.1109/CVPR.1994.323794>.
- Shi, W., Cao, J., Zhang, Q., Li, Y., Xu, L., 2016. Edge Computing: Vision and Challenges. IEEE Internet Things J. 3, 637–646. <https://doi.org/10.1109/JIOT.2016.2579198>.
- Snowden, C., Ritchie, G., Cave, J., Keeling, W., Rajan, N., 2013. Multiple irrigation levels affect boll distribution, yield, and fiber micronaire in cotton. Agron. J. 105, 1536–1544. <https://doi.org/10.2134/agronj2013.0084>.
- Sun, S., Li, C., Paterson, A.H., Chee, P.W., Robertson, J.S., 2019. Image processing algorithms for infield single cotton boll counting and yield prediction. Comput. Electron. Agric. 166, 104976 <https://doi.org/10.1016/j.compag.2019.104976>.
- Sun, S., Li, C., Chee, P.W., Paterson, A.H., Jiang, Y., Xu, R., Robertson, J.S., Adhikari, J., Shehzad, T., 2020. Three-dimensional photogrammetric mapping of cotton bolls in situ based on point cloud segmentation and clustering. ISPRS J. Photogramm. Remote Sens. 160, 195–207. <https://doi.org/10.1016/j.isprsjprs.2019.12.011>.
- Sun, M., Li, Y., Qi, Y., Zhou, H., Tian, L.X., 2022. Cotton boll localization method based on point annotation and multi-scale fusion. Front. Plant Sci. 13 <https://doi.org/10.3389/fpls.2022.960592>.
- Törnqvist, L., Vartiainen, P., Vartiainen, Y.O., 1985. How should relative changes be measured? Am. Stat. 39, 43–46. <https://doi.org/10.1080/00031305.1985.10479385>.
- Wallace, L., Hillman, S., Reinke, K., Hally, B., 2017. Non-destructive estimation of above-ground surface and near-surface biomass using 3D terrestrial remote sensing techniques. Methods Ecol. Evol. 8, 1607–1616. <https://doi.org/10.1111/2041-210X.12759>.
- Wang, L., Zhou, X., Zhu, X., Dong, Z., Guo, W., 2016. Estimation of biomass in wheat using random forest regression algorithm and remote sensing data. Crop J. 4, 212–219. <https://doi.org/10.1016/j.cj.2016.01.008>.
- Zaman-Allah, M., Vergara, O., Araus, J.L., Tarekegne, A., Magorokosho, C., Zarco-Tejada, P.J., Hornero, A., Albà, A.H., Das, B., Craufurd, P., Olsen, M., Prasanna, B.M., Cairns, J., 2015. Unmanned aerial platform-based multi-spectral imaging for field

- phenotyping of maize. *Plant Methods* 11, 1–10. <https://doi.org/10.1186/s13007-015-0078-2>.
- Zhang, Y., Yang, G., Liu, Y., Wang, C., Yin, Y., 2022a. An improved YOLO network for unopened cotton boll detection in the field. *J. Intell. Fuzzy Syst.* 42, 2193–2206. <https://doi.org/10.3233/JIFS-211514>.
- Zhang, Y., Yu, J., Chen, Y., Yang, W., Zhang, W., He, Y., 2022b. Real-time strawberry detection using deep neural networks on embedded system (rtsd-net): An edge AI application. *Comput. Electron. Agric.* 192, 106586 <https://doi.org/10.1016/j.compag.2021.106586>.