Article

Not peer-reviewed version

Deep Learning-Driven Weed Detection in Lettuce Farms: Box Annotation and Post-Segmentation

Shivang Parmar , Wenting Luo , Evan McGinnis , Kamel Didan * , Mark Siemens * , Haiquan Li *

Posted Date: 9 June 2025

doi: 10.20944/preprints202506.0642.v1

Keywords: Weed Detection; Lettuce; R-CNN; DETR; Segment Anything

 doi:10.20944/preprints202506.0642.v1

Article

Deep Learning-Driven Weed Detection in Lettuce Farms: Box Annotation and Post-Segmentation

Shivang Parmar [1,2], Wenting Luo [3,4], Evan McGinnis [3], Kamel Didan [3,*], Mark Siemens [3,*] and Haiquan Li [3,*]

[1] KEYS Program, University of Arizona, Tucson, AZ 85721, USA
[2] Gilbert Classical Academy, Gilbert, AZ 85234, USA
[3] Department of Biosystems Engineering, University of Arizona, AZ 85721, USA
[4] Statistics and Data Science Graduate Interdisciplinary Program, University of Arizona, Tucson, AZ 85721, USA
* Correspondence: haiquan@arizona.edu; Tel.: +1(520)621-1890

Abstract: Weed infestations cause billions of dollars in annual loss and devastate natural habitats. Current weed recognition methods remain vulnerable to seasonal and environmental variations, and their performance relies on tedious manual curation. To address these limitations, we proposed a straightforward framework that combined pre-trained deep learning models (including transformers) with simple box annotations and Segment Anything Model (SAM) for precise postprocessing boundary delineation. We evaluated this approach by comparing the state-of-the-art Faster R-CNN (Region-based Convolutional Neural Network) against the pioneering transformer-based DETR on lettuce-farm imagery. Of 939 annotated images, 760 (≈81%) were used for training, 92 (≈10%) for validation, and the remaining 87 (≈9%) reserved for independent testing. Faster R-CNN achieved an overall F1 score of 95.0%—97.5% for lettuce and 92.5% for weeds—while DETR achieved 87.1% overall, with 88.1% for lettuce and 86.1% for weeds. In both models, SAM achieved near-perfect segmentation—even for overlapping or closely spaced objects—by focusing on a single object per bounding box. This research not only automates weed detection to boost lettuce yield, but also enables targeted weeding application, reducing the treatment cost and environmental impact.

Keywords: Weed Detection; Lettuce; R-CNN; DETR; Segment Anything

1. Introduction

Agriculture is one of the most important sectors not only in the American economy but also in those of developing and low-income countries [1–3]. Lettuce constitutes a significant portion of this agricultural output in the United States. In 2022, lettuce accounted for nearly 20% of vegetables and melons sales revenue, totaling approximately $21.8 billion [4]. Romaine lettuce generated $1.54 billion; Iceberg lettuce generated $1.33 billion; and other lettuce generated $1.25 billion [4]. Beyond its economic value, lettuce is also an important dietary vegetable due to its nutritional benefits [5].

However, weed infestations hinder lettuce growth. During the seedling stage, weeds compete with lettuce for resources, reducing seedling vigor [6]. Weeds also harbor diseases and insects, which lower agricultural yields and damage natural habitats [6,7]. Furthermore, if weeds persist at harvest, they can infest subsequent crops [6]. Weed infestations are not only a problem in lettuce farming but have plagued agriculture broadly, costing the economy nearly $32 billion annually [7,8].

Various methods have been developed to mitigate the adverse effects of weeds. Manual weeding, in which workers remove weeds using basic tools or by hand, is effective for small lots. However, labor shortages constrain its availability [7,9]. Mechanical weeding employs equipment such as cultivators and weeders; it is more cost-effective but limited to specific crops. A universally applicable approach is chemical weed control, where herbicides are sprayed across fields to eliminate weeds. Nevertheless, these chemicals can contaminate soil and nearby water sources, occasionally

 doi:10.20944/preprints202506.0642.v1

reducing yields. Some herbicides are toxic to non-target organisms—including birds, fish, and insects—with negative ecological impacts [10]. Additionally, chemical use has adverse effects on human health and has driven the emergence of herbicide-resistant weeds [9].

Precision weed management—combining weed detection with selective herbicide application at targeted locations—offers a means to reduce the environmental and ecological impacts of blanket chemical treatments. Publicly available datasets like Weed25 have been developed to support the development and test of weed detection algorithms [11]. Specialized models targeting weeds in crops like polyhouse grown bell peppers [12] have also been created and evaluated [10]. These approaches employ a range of deep learning architectures, including R-CNN [13], YOLOv3 [14], YOLOv5 [15], and transformers-based models [16] (e.g., Swin transformer [17]). However, most studies require extensive dataset curation and segmentation annotations. In this study, we test a novel framework that omits segmentation during model learning, instead applying segmentation post hoc using the high-accuracy Segment Anything Model (SAM) [18]. We evaluated two representative box-based detectors (Faster R-CNN [19] and DETR [20]), for both weed detection and semantic segmentation. Faster R-CNN represents the state-of-the-art non-transformer object detector, whereas DETR is the pioneering transformer-based approach [21]. Pairing these detectors with SAM demonstrates the potential performance of similar algorithms for precision weed management.

2. Materials and Methods

The overview of the workflow is shown in Figure 1. The process begins with image acquisition, followed by manual object annotation using bounding boxes to label two categories: lettuce and weed. The annotated dataset is then divided into training, validation, and test sets. The training and validation subsets are used to independently train two object detection models: R-CNN and DETR. These trained models are evaluated on the test set using the ground-truth bounding boxes. Finally, the Segment Anything Model (SAM) is applied to generate precise object outlines within the predicted bounding boxes.

 doi:10.20944/preprints202506.0642.v1

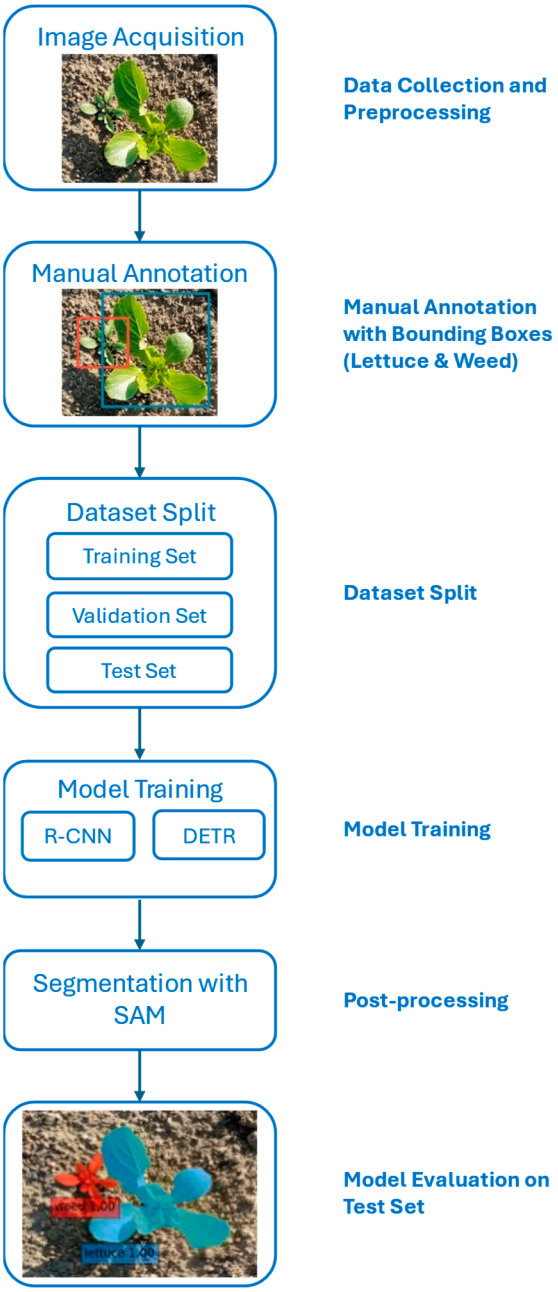


Figure 1. Overview of the Weed Recognition Workflow.

2.1. Data Collection and Preprocessing

To train both models, 939 close-up images of lettuce fields were captured by Samsung Galaxy S7 (SM-G930V). The images were collected at the University of Arizona Yuma Agricultural Center on January 18, 2019; November 17, 2020; and July 8, 2021, covering three seasons over three years, including various camera angles. The database includes two lettuce varieties: Iceberg and Romaine, spanning various growth stages. Each image originally measured 2160 x 2880 pixels; for model training, we resized them to 800 x 1333 pixels for optimal performance. Images with poor clarity or ambiguous content were excluded prior to annotation. Bounding boxes were annotated using LabelImg [22], with two classes: lettuce and weed. Annotations of images were exported in Pascal/Voc format [23], and was then converted to JSON [24] via custom Python scripts hosted on Github [25]. The images and associated annotations were then randomly partitioned into training (760 images), validation (92 images), and test (87 images) sets, following an approximate 8:1:1 split. Each set contained its respective images and a single consolidated annotation file. The same sets of

 doi:10.20944/preprints202506.0642.v1

data were used to train and evaluate Faster R-CNN and the transformer-based DETR algorithm to ensure a fair comparison.

2.2. Training (Faster R-CNN)

Detectron2 [26]—a PyTorch-based [27] deep learning library—was used to train our non-transformer models. Among the available models in Dectron2, we selected the Faster R-CNN (Region-based Convolutional Neural Network) [19] with a ResNet+ FPN backbone. The Faster R-CNN reused the features from convolution network for the classifier and proposal of regions, which later became a part of input features for the classifier. It optimized the extensive proposal stage significantly, compared with earlier versions such as R-CNN. We customized a pipeline from a RoboFlow [28] tutorial that demonstrates the use of Dectectron2 on a custom dataset [26,29]. In our implementation, images were subjected to horizontal and vertical flipping as needed, along with resizing and brightness adjustments. A few blurred images also underwent augmentation [29]. We used the same fine-tuning procedure as recommended in the Roboflow notebook, starting with default parameter values, including 1,000 maximum iterations, 300 warm-up iterations, and a batch size of 2 images. During training, we adjusted the parameters, such as exploring the learning rate between 0.0001 and 0.001, to further optimize model performance [29]. The pipeline was executed on both the Puma cluster of the HPC server and a local GPU server for reproducibility. The Puma cluster was equipped with a Nvidia V100 GPU, 512GB RAM, and ample storage. The local server was outfitted with an RTX 4090 GPU (24GB memory) and 256G RAM.

2.3. Training (Transformer)

We developed a customized DEtection TRansformer (DETR) [20] pipeline, drawing heavy inspiration from the Roboflow Google Collab tutorial notebook on training DETR with a custom dataset [30]. DETR is an end-to-end object detection model that combines convolution neuron networks with transformers, leveraging self-attention mechanisms [20]. The DETR model was fine-tuned from pre-trained models using a ResNet-50 backbone [31]. The default loss function, CrossEntropy, was employed as described in the original DETR paper [20]. During training, the learning rate and weight decay were set to 0.0001, while the learning rate for the backbone was set to 0.00001. The model was trained for 200 epochs. As with the Faster R-CNN model, the validation set of images was used for evaluating performance during training. All training was conducted on a local GPU server, as described in Section 2.3.

2.4. Post Segmentation with Segment Anything

After training either machine learning model, its predicted bounding boxes were individually fed into a Segment Anything [18] pipeline, which automatically identified the largest, typically centered, object within each bounding box and produced object masks. The Vision Transformer - Huge (ViT-H) model [32] was employed within SAM. The same server described in Section 2.3 was used to run the SAM model. It is important to note that SAM was applied using its default parameters without fine-tuning due to the broad generality of both the SAM algorithm and the associated models.

2.5. Testing

A coco evaluator was used to quantitatively assess both the Faster R-CNN and DETR models by calculating Average Recall (AR) and Average Precision (AP) on the test images, using a consistent Intersection over Union (IoU) threshold of 0.5. We customized the evaluation script from Roboflow's Google Collab notebook on Detectron2 and DETR [20]. As previously mentioned, these test images were randomly selected and included a variety of crop maturity stages and camera angles. It is important to note that only bounding box predictions were included in the quantitative evaluation, while segmentation performance was assessed through visual inspection only.

 doi:10.20944/preprints202506.0642.v1

3. Results

3.1. Overall Results

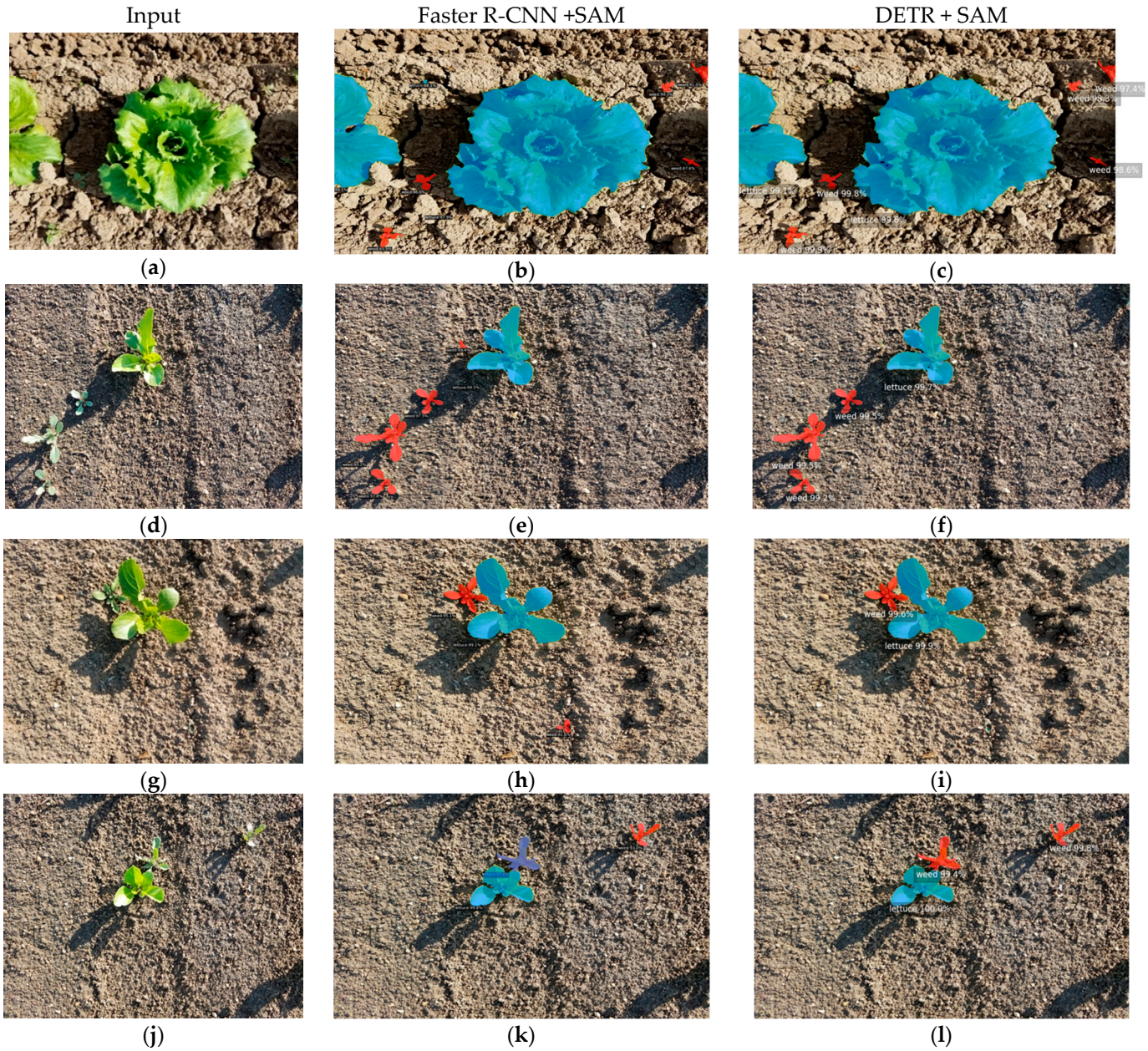
The overall performance of the testing dataset is summarized in Table 1. Faster R-CNN outperformed DETR, likely due to the limited dataset size, which can challenge transformer-based methods. Specifically, Faster R-CNN achieved an F_1 score of 97.5% (98.7% AR and 96.4% AP) for Lettuce and an F_1 score of 92.5% for weeds (96.2% AR and 89.1% AP). In contrast, DETR achieved an F_1 score of 88.1% (88.4% AR and 87.8% AP) for lettuce and 86.1% for weeds (87.5% AR and 84.8% AP). Pixel-level accuracy was not computed due to the absence of manually annotated segmentation masks for lettuces and weeds in our dataset. Instead, we assessed segmentation quality visually using the output masks derived from the predicted bounding boxes. Overall, the segmentation accuracy appeared high and nearly flawless. Only one segmentation error was encountered, marking soil as a weed region. Detailed segmentation results can be referred to some representative examples in Figure 2 or full test results provided in the Supplementary Material S1.

Table 1. Average recall (AR) and average precision (AP) scores of the Faster R-CNN and DETR models, expressed in percentage. Both scores are calculated by comparing the model's predictions to ground truth, using the Intersection over Union (IoU) of 0.5 as the cutoff for determining a match between predicted and true bounding boxes.

Method	$AR_{Combined}$	$AP_{Combined}$	$AR_{Lettuce}$	$AP_{Lettuce}$	AR_{Weed}	AP_{Weed}
Faster R-CNN	97.4	92.8	98.7	96.4	96.2	89.1
DETR	87.9	86.3	88.4	87.8	87.5	84.8

3.2. Case Studies of Weed Recognition in Typical Challenging Images

Figure 2 illustrates several typical challenging cases in which weeds are either close to or visually similar in shape to lettuce, across different stages of lettuce development and seasons. Overall, both Faster R-CNN and DETR successfully identified the main lettuce crops and nearby weeds, even though they were very close. Both algorithms appear to distinguish the crops and weeds based on shape and color, rather than being influenced by shadows.



(**m**) (**n**) (**o**)

Figure 2. Schemes follow the same formatting. Faster R-CNN and DETR, coupled with SAM, were used to make predictions excellently on the five example images taken in January 2019 and July 2021. The left columns show the original images; the middle columns show object detection results from Faster R-CNN + SAM; and the right columns show results from DETR + SAM. Blue masks indicate predicted lettuce, while red masks represent weeds.

 doi:10.20944/preprints202506.0642.v1

The first image (Figure 2a) shows a relatively simple example in which the lettuces are fully matured, resulting in differences in sizes and shapes between the lettuces and the weeds. Both Faster R-CNN (Figure 2b) and DETR (Figure 2c) correctly recognized one full lettuce, one partial lettuce, the two weeds between them, and three small weeds on the right. The segmentation results based on bounding box detection were highly accurate for both models, as seen from the masked outlines (Figure 2b and 2c) and the boundaries of the plants (Figure 2a).

In the second example, where lettuce and weeds are spaced apart, exhibiting similar shapes but different sizes (Figure 2d), both Faster R-CNN (Figure 2e) and DETR (Figure 2f) correctly identified the lettuce, three obvious weeds at the bottom left of the lettuce. In addition, Faster R-CNN correctly identified a small weed to the left of the lettuce (Figure 2e).

For the third case, where lettuce and weeds are in proximity, exhibiting similar shapes but different sizes (Figure 2g), both algorithms predicted the lettuce and a weed on the left. In addition, Faster R-CNN (Figure 2h) successfully identified a weed in the bottom right corner of the lettuce. Again, the segmentation results were highly accurate (Figure 2h and 2i).

In the fourth example, the shapes and sizes of two weeds closely resembled those of a small lettuce (Figure 2j). Both Faster R-CNN (Figure 2k) and DETR (Figure 2l) correctly identified the lettuce and the weed far apart from the lettuce, but only DETR identified the weed close to the lettuce.

In the last example, the lettuce was tiny with shapes that resembled those of weeds (Figure 2m). Faster R-CNN correctly identified all three lettuces (Figure 2n), but DETR mistakenly classified the top lettuce as a weed (Figure 2o).

In summary, both Faster R-CNN and DETR performed well in detecting the major objects and, additionally, were able to identify small weeds that are often overlooked by human curators.

4. Discussion

In this study, we aimed to establish a weed detection pipeline for crop fields without relying on labor-intensive manual segmentation. We used lettuce as a case study and employed Faster R-CNN and DETR to demonstrate the effectiveness of this approach. As shown by our overall performance metrics and challenging case studies, both models were able to accurately detect and generate bounding boxes of approximate size in most images. Despite the sample size being fewer than one thousand prior to augmentation, both models achieved promising results. Overall, the Faster R-CNN model achieved a higher average recall of 96-98% and a higher average precision score of 89-96%, compared to the DETR model, which had an average recall score of 87-88% and an average precision in the 84-87% range. Both models performed much better on large objects (e.g., nearly 100% recall for Faster R-CNN), but relatively much worse on very small lettuces that were visually like a weed. They also occasionally struggled to correctly separate objects of the same class when they were in proximity. Specifically, Faster R-CNN appeared to perform better on small weed recognition and identify many tiny weeds that were not originally annotated. Nevertheless, we speculate that with more data, transformer-based models like DETR could achieve performance comparable to Faster R-CNN.

For both models, SAM was used to create segmentations based on the predicted bounding boxes. As shown in the images, SAM correctly masked nearly all the objects detected. However, we observed that SAM struggled with a few small objects. It is important to note that SAM was not fine-tuned for this experiment. Therefore, with proper fine-tuning, these minor errors could likely be corrected, leading to improved overall performance.

Both R-CNN+SAM-based and DETR+SAM-based models achieved excellent results. With the addition of more training images, further improvements are expected in both models, particularly for the transformer-based DETR model, given its architectural advantage. It is important to note that the identification of lettuce as weeds (false positives for weeds and false negatives for lettuce) can potentially lead to permanent crop damage, although such cases are rare in our predictions. While there were some false negatives for weeds (i.e., undetected weeds), the extremely low rate of false positives for weeds (i.e., mistakenly classifying non-weeds as weeds) is remarkable. During weed

 doi:10.20944/preprints202506.0642.v1

removal, protecting lettuce from removal is critical, and due to the low incidence of lettuce being misidentified as weeds, this goal is largely achieved.

The research has significant implications. With early identification, herbicides or other intervention approaches (e.g., laser weeder) could be precisely applied to the masked region for each weed, limiting crop damage and minimizing environmental harm by reducing herbicide or laser usage [6,7,10]. Furthermore, reducing weed presence can lead to increased agricultural yield and lower production costs, as crops face less competition for resources—resulting in more efficient resource use [6]. As further work, we will focus on expanding the training dataset, fine-tuning SAM, and exploring more advanced models (e.g., Swin Transformer [17]) with higher sensitivity and scalability, to better detect weeds at its very early stage.

5. Conclusions

In conclusion, the study successfully demonstrated the effectiveness of using SAM in the post-analysis for weed detection in lettuce farms when coupled with both Faster R-CNN and DETR. Both models achieved high average precision scores and strong average recall scores, especially considering the sample size of less than one thousand images, although some challenges remain, such as false negatives for some small lettuce and inaccurate segmentation for tiny objects. Overall, the models performed well across varied camera angles, heights, different stages of lettuce growth, and multiple seasons. Although the Faster R-CNN model outperformed DETR in this study, we believe that with increasing training data and enhanced augmentation techniques, both models will eventually achieve comparable performance. The framework reduced the curation burden and advanced the mission of early detection and removal of weeds in their lifecycle, which is critical to reducing competition for resources and risks of hosting insects and diseases [6,7].

Supplementary Materials: The following supporting information can be downloaded at the website of this paper posted on Preprints.org. Supplementary Data S1: Original images, classification results, and segmentation results for the test set, obtained using Faster R-CNN and DETR, each coupled with SAM.

Author Contributions: Conceptualization: H.L., M.S., and K.D.; methodology: H.L., S.P., W.L., and E.M.; software: S.P., H.L., and W.L.; validation: H.L. and S.P.; formal analysis: S.P.; investigation: H.L., K.D., M.S.; resources: M.S.; data curation: S.P. and M.S.; original draft preparation: S.P.; review and editing: H.L. and S.P.; visualization: S.P. and H.L.; supervision: H.L.; project administration: H.L.; funding acquisition: H.L. All authors have read and agreed to the published version of the manuscript.

Funding: The study was partially supported by the BIO5 KEYS Research Internship Program, a BIO5 seed grant, and a startup package from the College of Agriculture, Environmental and Life Science.

Data Availability Statement: The dataset will be available at github.com/haiquanua

Acknowledgments: The authors appreciate the anonymous reviewers and early efforts on Detectron2 from Breeze Scott and Jacky Cadogan.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

R-CNN	Region-based Convolutional Neural Network
DETR	Detection Transformer
SAM	Segment Anything Model
AR	Average Recall
AP	Average Precision

 doi:10.20944/preprints202506.0642.v1

References

1. Cervantes-Godoy, D.; Dewbre, J. Economic importance of agriculture for poverty reduction. **2010**.
2. Beckman, J.; Countryman, A.M. The importance of agriculture in the economy: impacts from COVID-19. *American journal of agricultural economics* **2021**, *103*, 1595-1611.
3. Newman, C.; Singhal, S.; Tarp, F. Introduction to understanding agricultural development and change: Learning from Vietnam. **2020**, *94*, 101930.
4. USDA Economic Research Service. U.S. lettuce production shifts regionally by season. **2023**.
5. Kim, M.J.; Moon, Y.; Tou, J.C.; Mou, B.; Waterland, N.L. Nutritional value, bioactive compounds and health benefits of lettuce (Lactuca sativa L.). *Journal of Food Composition and Analysis* **2016**, *49*, 19-34.
6. Harker, K.N.; O'Donovan, J.T. Recent weed control, weed management, and integrated weed management. *Weed Technology* **2013**, *27*, 1-11.
7. Kubiak, A.; Wolna-Maruwka, A.; Niewiadomska, A.; Pilarska, A.A. The problem of weed infestation of agricultural plantations vs. the assumptions of the European biodiversity strategy. *Agronomy* **2022**, *12*, 1808.
8. Beck, L.; Wanstall, J. Noxious and troublesome weeds of New Mexico. *New Mexico State University, College of Agriculture, Consumer and Environmental Sciences, Las Cruces* **2021**.
9. Abbas, T.; Zahir, Z.A.; Naveed, M.; Kremer, R.J. Limitations of existing weed control practices necessitate development of alternative techniques based on biological approaches. *Advances in Agronomy* **2018**, *147*, 239-280.
10. Liu, B.; Bruch, R. Weed detection for selective spraying: a review. *Current Robotics Reports* **2020**, *1*, 19-26.
11. Wang, P.; Tang, Y.; Luo, F.; Wang, L.; Li, C.; Niu, Q.; Li, H. Weed25: A deep learning dataset for weed identification. *Frontiers in Plant Science* **2022**, *13*, 1053329.
12. Subeesh, A.; Bhole, S.; Singh, K.; Chandel, N.S.; Rajwade, Y.A.; Rao, K.; Kumar, S.; Jat, D. Deep convolutional neural network models for weed detection in polyhouse grown bell peppers. *Artificial Intelligence in Agriculture* **2022**, *6*, 47-54.
13. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2014; pp. 580-587.
14. Farhadi, A.; Redmon, J. Yolov3: An incremental improvement. In Proceedings of the Computer vision and pattern recognition, 2018; pp. 1-6.
15. Jocher, G.; Stoken, A.; Borovec, J.; Changyu, L.; Hogan, A.; Diaconu, L.; Ingham, F.; Poznanski, J.; Fang, J.; Yu, L. ultralytics/yolov5: v3. 1-bug fixes and performance improvements. *Zenodo* **2020**.
16. Jiang, K.; Afzaal, U.; Lee, J. Transformer-based weed segmentation for grass management. *Sensors* **2022**, *23*, 65.
17. Wang, Y.; Zhang, S.; Dai, B.; Yang, S.; Song, H. Fine-grained weed recognition using Swin Transformer and two-stage transfer learning. *Frontiers in Plant Science* **2023**, *14*, 1134932.
18. Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A.C.; Lo, W.-Y. Segment anything. In Proceedings of the Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023; pp. 4015-4026.
19. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence* **2016**, *39*, 1137-1149.
20. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In Proceedings of the European conference on computer vision, 2020; pp. 213-229.
21. Vaswani, A. Attention is all you need. *Advances in Neural Information Processing Systems* **2017**.
22. Tzutalin, D. Labelimg. git code. Available online: https://github.com/tzutalin/labelImg (accessed on
23. Everingham, M.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes (voc) challenge. *International journal of computer vision* **2010**, *88*, 303-338.
24. Pezoa, F.; Reutter, J.L.; Suarez, F.; Ugarte, M.; Vrgoč, D. Foundations of JSON schema. In Proceedings of the Proceedings of the 25th international conference on World Wide Web, 2016; pp. 263-273.
25. Spinellis, D. Git. *IEEE software* **2012**, *29*, 100-101.
26. Wu, Y.; Kirillov, A.; Massa, F.; Lo, W.-Y.; Girshick, R. Detectron2. **2019**.

 doi:10.20944/preprints202506.0642.v1

27. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* **2019**, *32*.
28. Alexandrova, S.; Tatlock, Z.; Cakmak, M. RoboFlow: A flow-based visual programming language for mobile manipulation tasks. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), 2015; pp. 5537-5544.
29. Solawetz, J. How to Train Detectron2 on Custom Object Detection Data. *Roboflow* **2021**.
30. Skalski, P. *How to Train RT-DETR on a Custom Dataset with Transformers* **2024**, *2024*.
31. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016; pp. 770-778.
32. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* **2020**.