Detecting weeds using computer vision and deep neural networks

Abstract

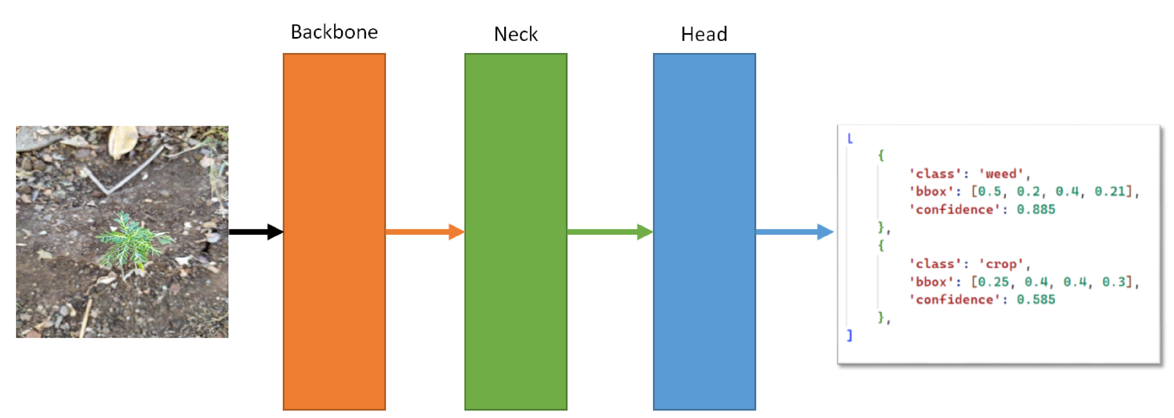
Weed control is an important topic in agriculture, and efforts are continuously made to enhance the automation of this process. Using **computer vision** we can employ intelligent agents to accurately detect weeds as a first step in weed control.
By training a **convolutional neural network**, we can let the computer extract relevant features itself to tell apart harmful weeds from agricultural crops.
We used the state-of-the-art **Yolov8** CNN, fine tuned on a public dataset featuring images of crops and weeds, labeled into two classes and annotated with bounding boxes.
The model proved to be effective, reaching a F1 score above 0,8 while being able to evaluate images relatively quickly.
A larger, better annotated and more balanced dataset could further improve the model.

Introduction

Weeds are one of the largest problems in agriculture today. The cause of weeds are: monoculture production farming, using low quality seeds, poor soil, parcels that are too large or too small for the crop that's grown there and using bad fertilizers.
Case study: **wheat** is the 2nd largest culture in Serbia, with as much as 24,3% of agricultural soil being wheat fields. There have been over 200 species of weeds attacking wheat recorded as of 2022.
To maintain crop quality and maximize income, as well as keep the soil healthy, it is necessary to undergo **weed control** – monitoring and removal of weeds. There are several methods of weed removal, most notably pulling weeds and using weedicides.
Many farms around the world still rely on manual labor for weed control. With the advent of **computer vision**, there has never been a better time to move towards automation.
This project attempts to bring forth one such solution. By utilizing a **convolutional neural network** trained on tailor-made data, we can automate the otherwise difficult job of culling weeds from crops.

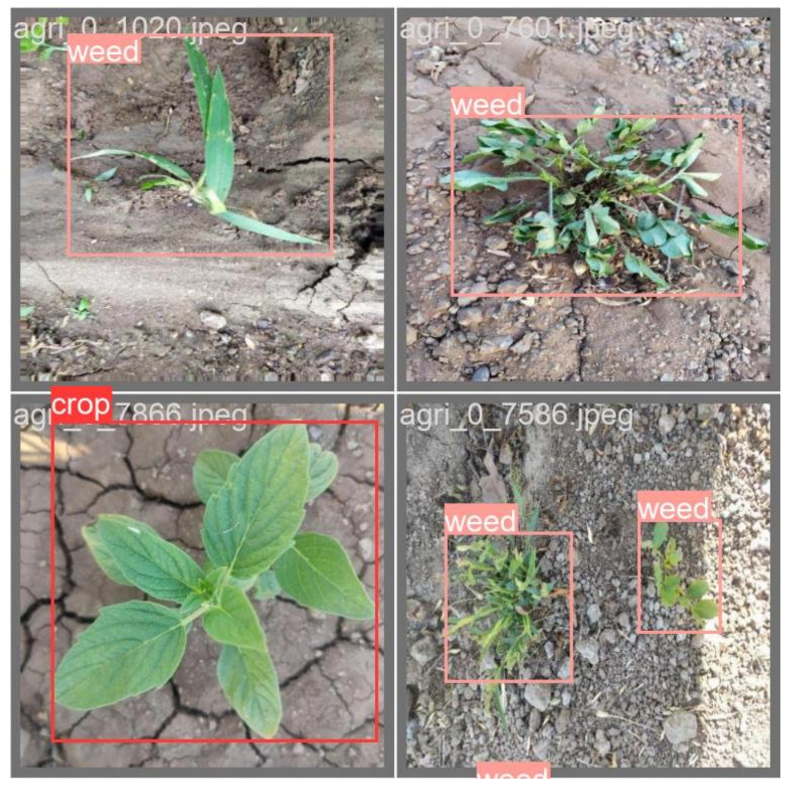
Methodology

We shall use **Yolov8**, a state-of-the-art object detection CNN.
The Yolov8 architecture has three parts. The **backbone** extracts features using convolution layers, batch normalization and spatial pyramid pooling.



The **neck** combines the extracted features by concatenating and upsampling feature maps.
The **head** performs twofold prediction: one prediction for object detection (bounding boxes) and one class prediction.

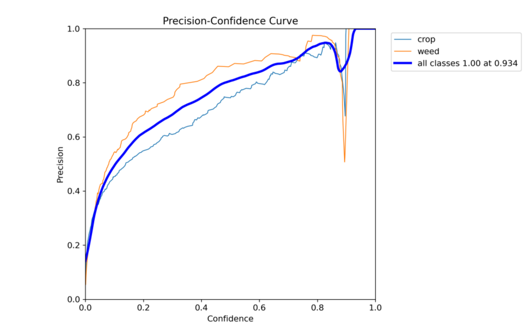
Our model will be trained on a publicly-available weeds dataset of 1300 images of sizes 512x512. Each image is a close-up of one or many crops and weeds on brown soil. There are **two classes** in the dataset: *crop* and *weed*. Each object has a bounding box and a label for one of the classes it belongs to.

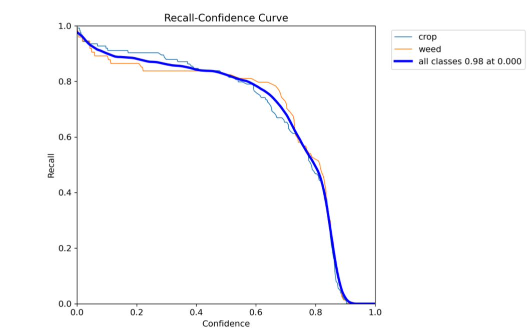


The dataset is relatively small, so training the CNN from scratch would produce unsatisfactory results. Instead, we **fine tune** a pre-trained YOLOv8 model (trained on the COCO dataset).
Our dataset is split into a training, validation and test set with a ratio of 80:10:10. The model trained for 100 epochs.

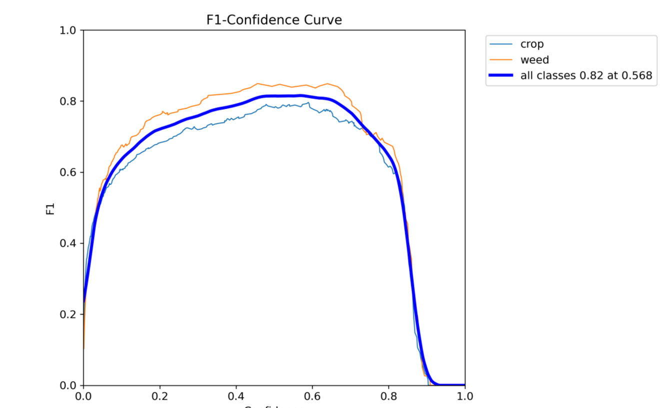
Results

During training, the box loss dropped from 1,4 to 0,8. Validation yielded worse results, from 1,6 to 1,5. As for class loss, the training set went from 2,2 to 0,5 while the validation set went from 2,4 to 0,8.

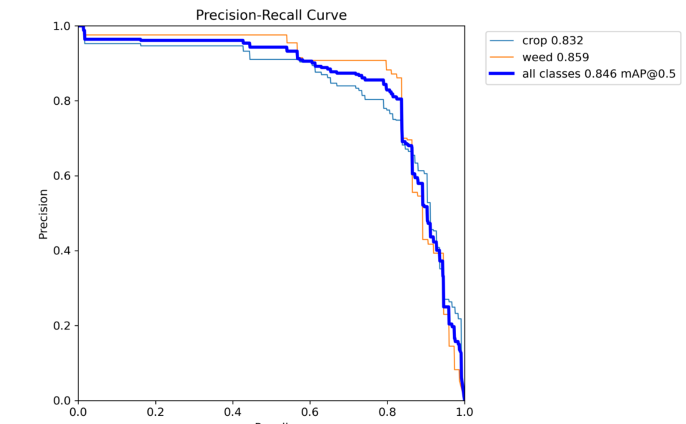




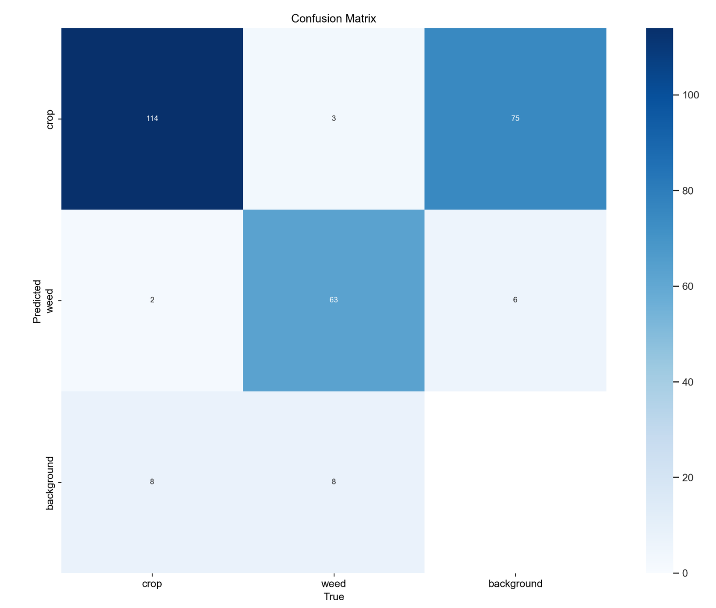
We got a **precision** of 0,822 and a **recall** of 0,805. With an **F1 metric** of 0,813, our model has a decent balance of precision and recall.



Our model reached an **average precision** of 0.83.

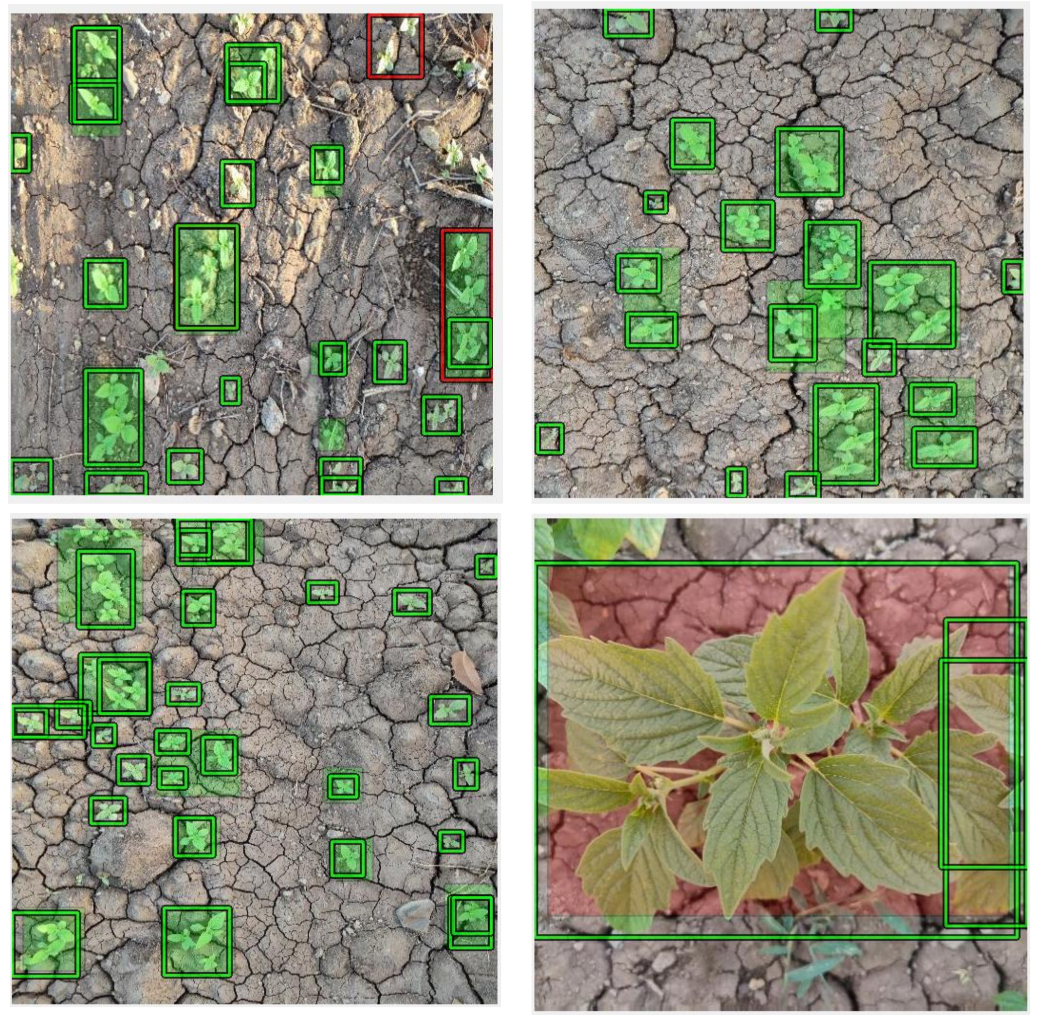


As for **mean average precision**, we will take into consideration two cases: one with an IoU threshold of <50, and one for an IoU threshold between 50 and 95. The first case has a mAP of 0.846, while the second one is only 0.584.
There is a high recall for the 'crop' class being detected under a background, as can be seen from the confusion matrix.



Conclusion

Some of the inaccuracies can be attributed to **poor annotation** of the dataset. Our model was able to detect crops and weeds that were only partially visible, as well as small objects that weren't annotated in the dataset.
On the other hand, our model sometimes **incorrectly classifies** crops as weeds. For later use of the model, we would prefer maximizing classification metrics like F1 over detection metrics like IoU.



The image above shows a sample of the test results. Outlined are the model's predictions while the transparent rectangles are ground truths. Crops are colored in green, while weeds are colored in red.

The dataset is **not balanced**. There are more images of crops (as well as more crop objects in general) than there are of weeds, therefore it is possible that our model is **biased** towards crops.
Performance-wise, the model can do prediction in ~1 second per image, which is close to being applicable for real-time environments. We could reduce prediction time by using smaller images, though this may result in a lower recall.

References

Agriculture report in Serbia 2022

Summary of the Yolov8 architecture

Crop and weed dataset