Project H1: KAGGLE - The 3LC Cotton Weed Detection Challenge

Siim Ilison and Markus Maripuu
GitHub

Business understanding

Background

The 3LC Cotton Weed Detection Challenge on Kaggle is a competition that aims to automatically detect weeds in cotton fields based on images. Weed control is one of the most labour-intensive and costly tasks in cotton farming. Manual weed identification requires experienced workers and takes significant time, while chemical overuse is expensive and environmentally harmful. An accurate object detection model would enable farmers and agronomists to automate early weed detection, optimize pesticide usage, and potentially support autonomous agricultural machinery.

Business Goals

Our project aims to build a reliable object detection model that can locate and classify 3 different weeds in cotton fields from imagery. The direct business goal is to reduce manual labour, lower pesticide expenditure, and improve crop health.

Business Success Criteria

- The resulting model should correctly detect weeds with a high enough accuracy to make practical decisions.
- It should reduce the amount of manual inspection needed.
- The model's predictions should be interpretable and visually inspectable.
- From a course perspective: the project should fulfil CRISP-DM requirements, demonstrate proper methodology and achieve a meaningful baseline or competitive score on Kaggle's leaderboard.

Inventory of Resources

- Annotated dataset from the Kaggle 3LC Cotton Weed Detection Challenge (images + bounding-box annotations).
- GPU compute through Kaggle Notebooks and CPU compute on our own computers
- YOLOv8n / Ultralytics library for training detection models.
- Python libraries (PyTorch, NumPy, pandas, matplotlib, 3LC).YOLOv8n / Ultralytics library for training detection models.
- 3LC for visual analysis and data corrections
- GitHub repository
- Roboflow for data augmentation

Requirements, Assumptions, and Constraints

- We assume that the dataset is representative of typical cotton fields.
- The project must be completed within the course deadline and follow the CRISP-DM methodology.
- The only allowed model is YOLOv8n.
- The final deliverable must be reproducible via the project repository.
- The competition dataset may contain noisy labels or inconsistent annotation quality, which constrains maximum possible accuracy.

Risks and Contingencies

- **Insufficient model performance** → We will apply data augmentation, hyperparameter tuning.
- **Annotation inconsistencies** → Perform exploratory data quality checks in 3LC
- **Time constraints** → Training the model takes a lot of time so we will run it simultaneously on two computers if needed
- **Incorrect data labeling** → Modifying bounding boxes and missed labels on the base data (train and validation dataset).

Terminology

- **Bounding box**: a rectangular annotation around each weed instance.
- **Object detection**: locating and classifying objects in images.
- **Precision/Recall**: metrics evaluating model correctness and coverage.
- **mAP50 (mean Average Precision)**: main performance metric in detection tasks and the metric used for this Kaggle competition.

Costs and Benefits

Costs are limited to team time, compute resources, and development effort. Benefits include producing a useful model for precision agriculture, gaining applied ML experience, and producing a strong course project.

Data-Mining Goals

- Train an object detection model to detect weeds in cotton fields.
- Produce high-quality predictions for the Kaggle submission format.
- Explore dataset patterns, label types, and image properties.
- Evaluate models based on mAP and qualitative inspection.

Data-Mining Success Criteria

- Achieving a model with stable training and reasonable mAP on validation data.
- Producing correct submission files accepted by Kaggle.
- Clear documentation of workflow and results in the project report and repository.

Data Understanding

Gathering Data

Outline Data Requirements

To train an object detection system, we require:

- RGB images of cotton fields.
- Bounding-box annotations marking individual weeds.
- Metadata describing image size and label structure.
- Clear definition of the prediction target (weed locations + class labels).

These requirements derive directly from YOLO-style object detection pipelines.

Verify Data Availability

All needed data is provided by the Kaggle competition:

- **train/** directory containing images and labels for training
- **val/** directory containing images and labels for validation
- **test/** directory for unlabeled prediction images
- The dataset is complete, publicly accessible, and already split into training/validation/test sets.

Define Selection Criteria

We will use:

- The full training set for model training, except possible removal of duplicates or empty-label images.
- The full validation set for testing the model before generating Kaggle predictions.
- Images resized to a consistent training resolution (640×640).

The test set will be used only for generating predictions for Kaggle. 50% of the test dataset is in the public split and will be used for the public leaderboard before the competition ends.

The other 50% is in the private split and will be used to decide the best models predictions of the whole competition.

Describing Data

Images

- Resolution varies; most images are high-resolution overhead crop images.
- Images show different plants, soil background, and weed patches.
- The weed is not always in the middle of the image
- The colour distribution is natural RGB, often bright with soil-plant contrast.
- The weeds from various pictures are overlapping

Labels

Each label file corresponds to a single image and contains lines in the format:

```
class_id x_center y_center width height
```

where all values are normalized to [0,1].
The challenge involves **three main target classes: Carpetweed, Morning Glory, Palmer Amaranth**

Dataset Structure

- 542 training images and label files
- 133 validation images and label files
- 170 testing images

Exploring Data

We will explore the dataset in 3LC using:

- **Class distribution**: verify whether weeds appear uniformly or some images contain none.
- **Bounding box sizes and shapes**: inspect how well the bounding boxes fit the weeds.
- **Image brightness and contrast**: check whether lighting conditions vary heavily.
- **Distribution of weed counts per image**: e.g., are many weeds clustered or isolated?

Observations from 3LC show that:

- Weeds are often mislabelled or labelled multiple times
- Some images are visually noisy (shadows, soil texture).
- Bounding boxes vary significantly and often include overlapping vegetation.

Verifying Data Quality

Data quality checks include:

- **Check for and fix annotation errors** using 3LC.
- **Verify consistency of bounding box normalization**.

Based on inspection, the dataset is very poorly labelled and almost every image requires:

- consistent resizing
- ensuring the correct label

Project plan

Task	Person responsible	Time
Importing all needed packages and libraries	Both	2h each
Learning how to use 3lc and run the model	Both	4h each
Exploring data and trying to understand the main goal and how are we going to achieve that	Both	3h each
Experimenting and getting to know to YOLOv8n model	Both	1h each
Fixing incorrect bounding boxes and labels	Both	10h each
Tuning hyperparameters of the YOLOv8n model	Both	6h each
Getting the model to work on Kaggle GPU compute	Both	2h each
Training the model with different hyperparameters	Both	1h each
Data augmentation using Roboflow	Both	1h each
Results analysis	Both	3h each
Writing reports	Both	2h each
Creating and fulfilling requirements for the GitHub repository	Both	2h each
Creating the poster	Both	4h each