Germination CV Assesment

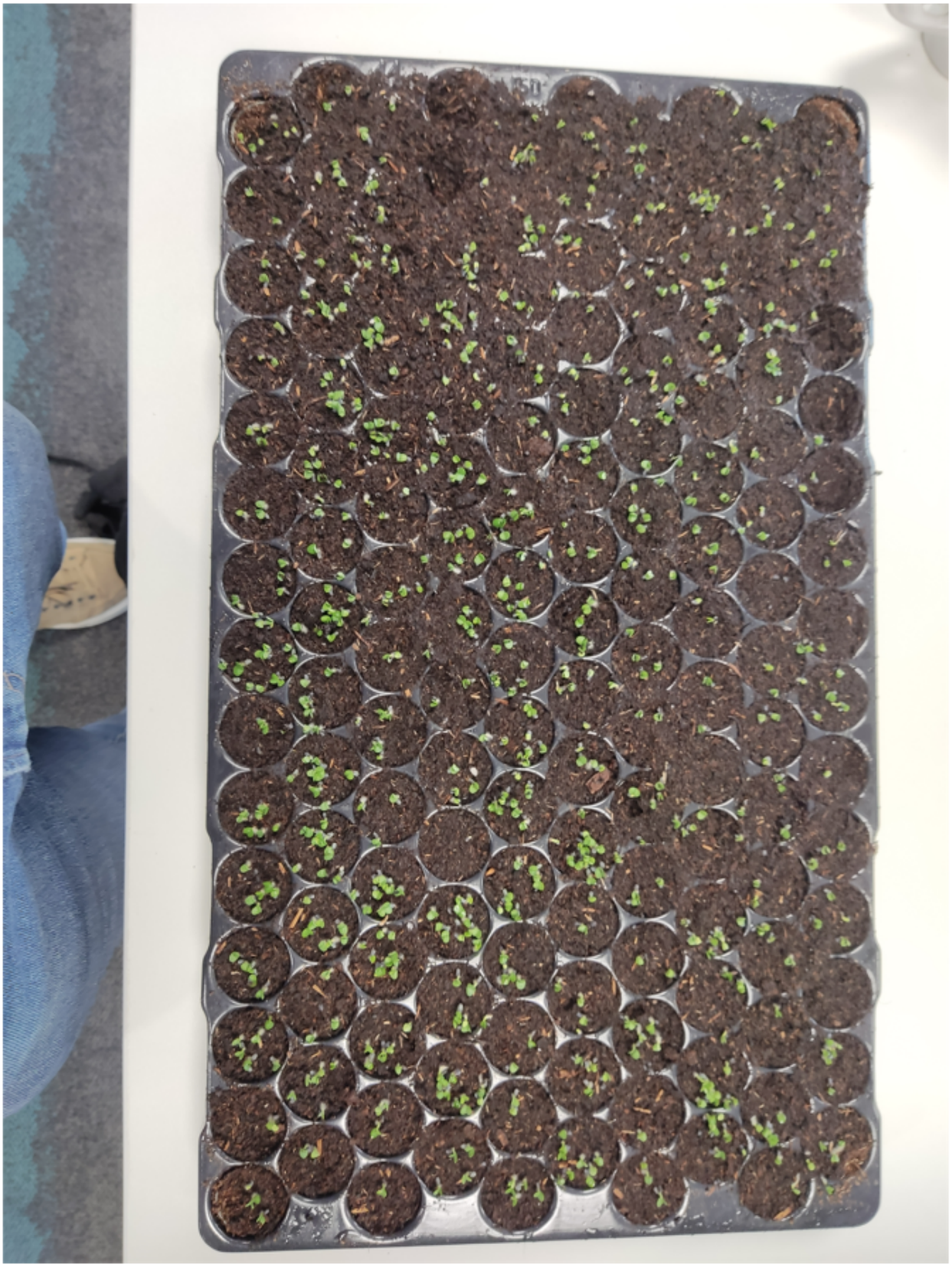
Created By	
Stakeholders	
Status	Approved
Type	Project report
Created	@April 12, 2022 1:56 PM
Last Edited Time	@May 4, 2022 9:29 AM
Last Edited By	

Germination CV Assesment

In this project we used a camera image of a germination tray with basil seedlings, which are kept in a seperate germination chamber, to asses which seedling plugs are best for transplanting into the Vertical Farm(VF). To ensure sufficient seedlings grow the plugs are sewn excessivly. Plugs that do not atleast contain 5 seedlings are considered not suitable for the VF but a too large amount will also prevent proper growth. Furthermore the farther the seedlings are seperated the better they grow. The goal is to use CV techniques to identify these 2 metrics for every plug.

Dataset

The dataset for this particular project actually only contained a single image of a single tray which is shown below.

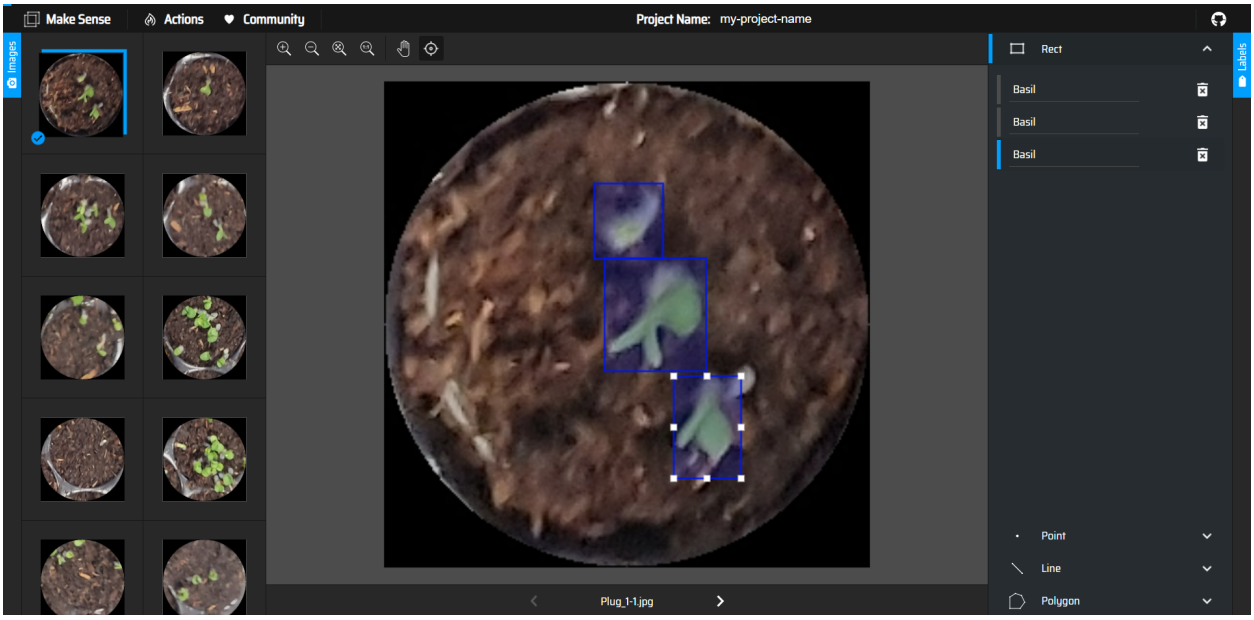


This single tray contains 150 plugs with varying amount and spread of seedlings. Therefore by assesing every plug individually you can use this image as 150 datapoints. To create these individual images of the plugs we use the openCV python library. Using this we drew a circle on bottom left plug. Then by cutting out a bounding box around it we created the following image.

Then from this first circle we used a linear pattern to cut out all the other plugs from the image. As the tray image was not perfectly photographed from above, it is very difficult to fit all plugs perfectly with a linear pattern. This leads to not all plugs being fully covered, but for the purpose of this project it is sufficent.

Labeling data

As this is a object detection problem, to train a model the images need to be labeled. To do so we used a browser application called MakesenseAI. On this website you can upload images, which are then easily lableable as can be seen here.



For these images we labeled all the individual seedlings as ‘Basil’ and if the image showed no seedlings it was labled as ‘Nothing’, since all images require a label. This way all 150 images were labeled and used as input for the machine learning model.

Model training

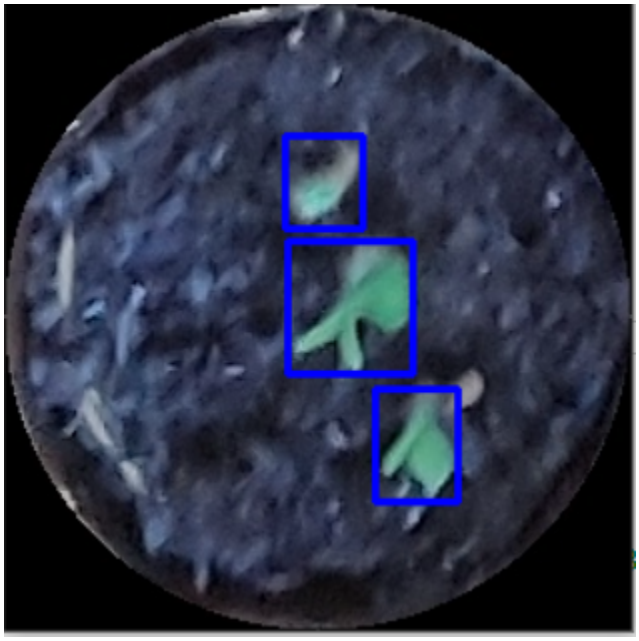
To train a ML model for this specific object detection CV problem, we used a python library Detecto, which is based on pytorch. This library has a set of useful functions for object detection problems along with a structure for a convolution neural network(CNN). With the following simple lines of code and this library an object detection model can be trained.

```
Train_dataset=core.Dataset('../Data/Train_set/')#L1
Test_dataset = core.Dataset('../Data/Test_set/')#L2
loader=core.DataLoader(Train_dataset, batch_size=2, shuffle=True)#L3
model = core.Model(['Basil', 'Nothing'])#L4
losses = model.fit(loader, Test_dataset, epochs=10, lr_step_size=5, learning_rate=0.001, verbose=True)#L5
model.save('model_weights.pth')
```

In this model we used a train test split of 120/30 (80%/20%) and 10 epochs of training time.

Model performance

After creating this model we tested it on image 1-1 to see how it works.



This shows that the model produces bounding boxes on what it percieves to be seedlings, which in this perticular case is done quite accurately.

Model Validation

To validate the model we only consider the amount of seedlings detected as validating the bounding boxes is quite complex and is also less important than the seedling amount. For this purpose we calculated the Mean Absolute Error(MAE) for the test set of 30 images which was 0.233.