High-Throughput Phenotyping using Computer Vision and Machine Learning

Vivaan Singhvi, Langalibalele Lunga, Pragya Nidhi, Chris Keum, Varrun Prakash

Farragut High School

Introduction and Background Information

High-throughput phenotyping is a breakthrough technology used in plant biology and agriculture to examine a wide variety of plant features through **image analysis**. By coupling this technique with **machine learning**, scientists can handle plant datasets **thousands of times larger** in a **fraction of the time**.

Research Objective

Using a dataset of **1672** images (with spanning **white labels**) of the plant **Populus Trichocarpa**, provided by Oak Ridge National Laboratory, we aim to address the following summarized challenge questions:

1. Is it possible to use **optical character recognition** to "read" each label and **generate a spreadsheet** of the features on the label?
2. Can **machine learning** classify different **leaf morphologies** among plants, such as leaf shape or color?
3. Can a **predictive model** be built **using leaf morphology classifications** that can indicate the **condition** in which a plant was raised?
4. GPS and other camera information are encoded in **EXIF tags**. Can this data be used to determine characteristics such as leaf size? Can other data, such as soil maps, weather, etc. be used to find **correlations among phenotypes**?

Reading Labels with Optical Character Recognition

- The prebuilt model **PaddleOCR** was chosen for label reading due to its **accuracy** and **efficiency**.
- However, minor **image augmentation** was needed to configure images, including **edge highlighting** and **rotating**.

Figure 1. The image augmentation process for optical character recognition

- **Regular Expressions** extracted features from text.
- On 30 random images, the model had an accuracy of **77.33%**, **94.31%** with null values omitted.

filename	treatment	block	row	position	genotype
...	D	1	8	32	BESC-34
...	C	1	10	12	**BESC-417_LM**,core
...	C	2	3	40	BESC-468
...	C	2	6	54	BESC-28_LM
...	C	1	24	22	**LILD-26-5_LM**,core
...					**HOMD-21-2_LM**,core
...	C	2	23	45	BESC-361_16_11_CB
...		1	25	30	**BESC-106_LM**,core

Table 1. Example spreadsheet data

Classifying Leaf Morphologies with Image Segmentation

- The **Segment Anything Model** (SAM) was chosen to segment leaves. The `SamPredictor` could generate an exact mask at a **given point**.
- To approximate these points, the below pipeline was used, using **HSV Filtering**, **Edge Detection**, and **Dilation** techniques.

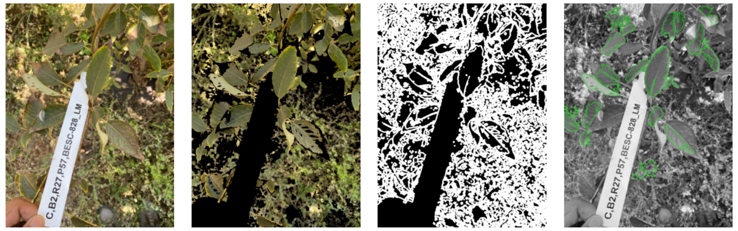

Figure 2. The image processing pipeline utilized to generate leaf approximations

Figure 3. Segmentation masks after filtering

- The **centers** of these approximations were used as the **generation point**.
- Then, using a **machine learning classifier**, we **filtered bad segmentations** (shown in red to the left) with an accuracy of **90.91%**.
- The masks are then used to **classify morphologies**.
- The chosen features were **color**, **shape**, and **level of brown splotch** (indicating withering leaves).
- The `XGBoost` model classified the features, and `scikit-learn`'s `MultiOutputClassifier` was implemented to manage all three at once.

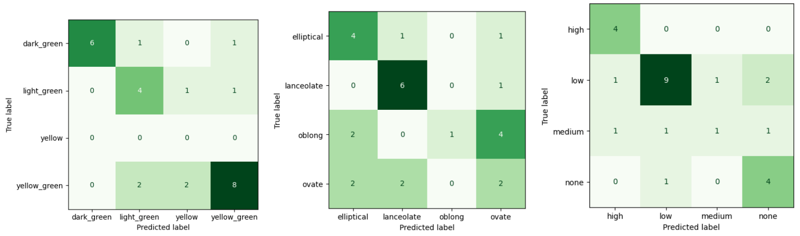

Figure 4. Confusion matrices for color, shape, and splotch respectively

- The **color** classifier had an accuracy of **69.23%**, and seemed to be confused around **true yellow-green** leaves.
- The **shape** classifier was more sporadic with an accuracy of **50.00%**, **mispredicting oblong** and **ovate** leaves the most.
- The **splotch** classifier had an accuracy of **69.23%**, without significant errors.
- The model was used on **every image** by finding the **mode prediction** for each feature. Then, this data was **inserted into the spreadsheet**.

Predicting Treatment from Morphological Classifications

- By using **read treatments** from step 1 in conjunction with **morphological classifications** from step 2, we could build a simple predictor to determine if a plant was raised in **drought** or **control**.
- **One-hot encoding**, as seen to the right, was used to convert our **qualitative** data to **quantitative** data.

leaf_color	light_green	dark_green	yellow_green	yellow
light_green	1	0	0	0
yellow_green	0	0	1	0
dark_green	0	1	0	0
light_green	1	0	0	0
yellow	0	0	0	1

Table 2. Example of one-hot encoding

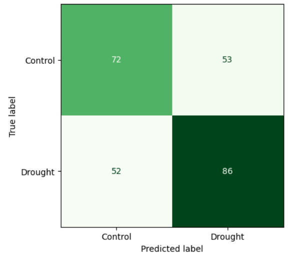

Figure 5. Confusion matrix for treatment classifier

- Some data had to be **pruned** as to avoid **class imbalance**.
- Our model, a `RandomForestClassifier`, had an accuracy of **60.08%** and a confusion matrix shown on the left.
- The low accuracy is likely due to our limitation of **only using classification data** from the previous step. Along with the fact that they **may be inaccurate**, only three features are **likely not enough** to make good predictions.

Finding Correlations and Characteristics from EXIF Tags

- The EXIF tags were **not useable** for predicting leaf size, or other traits.
- A vital tag, the `FocalPlaneResolution`, was **missing** from the images.
- Additionally, since all leaves were close together, **no weather or soil map API** would provide geolocational data **specific enough** for us.
- If this information were **present in a given file**, we could make conclusions.

Conclusion and Significance

- We were **successfully** read plant labels with **optical character recognition** and store the data in our spreadsheet.
- We were **successfully** able to extract leaves from the images and **somewhat successfully** able to classify their **morphologies**.
- We were **slightly successfully** able to determine **treatment** from morphology classifications and **not successful** in making conclusions from **EXIF tags**.
- Regarding **originality**, this study is **one of the first** to implement **PaddleOCR** and the **Segment Anything Model** in this context, due to their relative recency. Their powerful capabilities proved to be **vital** to our research.