



PhytoOracle Phenomics Data Processing Pipelines*

From Field Preparation to Phenotype Information

Emmanuel Miguel Gonzalez, Jeffrey Demieville, Sebastian Calleja,
Bryan Pastor, Rodrigo Silva, Duke Pauli

2024-01-24

*The PhytoOracle project is supported by the following grants: U.S. Department of Energy Biological and Environmental Research (DE-SC0020401) and Advanced Research Projects Agency - Energy OPEN (DE-AR0001101); National Science Foundation Plant Genome Research Program (IOS-2102120, IOS-2023310, and IOS-1849708), Division of Biological Infrastructure (2019674 and 1743442), and CyVerse project (DBI-1743442); Cotton Incorporated (18-384, 20-720, 21-830, and 23-890); and U.S. Department of Agriculture National Institute of Food and Agriculture Specialty Crop Research Initiative (2021-51181-35903).

Table of Contents

1 Field Preparation	4
1.1 Planting	4
1.1.1 Equipment	4
1.1.2 Potential Issues During Planting	7
1.2 Ground Control Points	7
1.3 Thinning	9
2 Positioning Information Preparation	10
2.1 Collecting Global Positioning System (GPS) Coordinates	10
2.2 Positioning Information Files Required by PhytoOracle	12
2.2.1 Generating GCP File	12
2.2.2 Generating GeoJSON File	12
3 Pipeline Preparation	15
3.1 Editing YAML file	15
3.2 Updating GitHub Repositories	15
3.2.1 PhytoOracle Data	15
3.2.2 PhytoOracle Landmark Selection	16
4 Intro to High Performance Computers	19
4.1 UArizona High Performance Computing Cluster	19
4.1.1 Compute System	19
4.1.2 Compute Resources	19
4.1.3 Further Information	20
5 Running PhytoOracle on High Performance Computers	20
5.1 Defining Compute Resources	20
5.2 Before Deploying PhytoOracle	21
5.3 Supported Data Types	21
5.3.1 2D Field Scanner Data Types	21
5.3.2 3D Field Scanner Data Types	21
5.4 Deploying PhytoOracle	21

5.4.1	stereoTop	26
5.4.2	flirIrCamera	26
5.4.3	ps2Top	27
5.4.4	scanner3DTop	27
5.5	Quality Control & Quality Assurance of Processed Data	28
6	Data Visualization	29
6.1	PhytoOracle Dashboard	29

1 Field Preparation

This section details field preparation for lettuce seasons. Similar, but different, steps are followed for sorghum seasons. Driving factors for field management practices are agronomic requirements, technical requirements defined by the Maricopa Field Scanner, and requirements for automated and manual phenotyping workflows.

The following steps must be completed prior to planting:

1. Shape raised beds
2. Inject subsurface drip irrigation tape
3. Reshape beds
4. Mark the seed line
5. Set up sprinkler irrigation, including pipes, heads, gaskets, and filters
6. Place string, metal spikes and labeled stakes in the field

Note: These steps are carried out by Pauli Lab members a few weeks before planting.

After completing these steps, the field will look like [Figure 1](#).

1.1 Planting

Lettuce planting generally occurs around Mid-November to early-December. The mean air temperature during this time has previously ranged from 10 °C to 22 °C ([Figure 2](#)).

1.1.1 Equipment

Planting is done by hand using [Earthway garden seed planters](#) ([Figure 3 Left](#)). Lettuce seeds must be planted at a depth of 1/8 to 1/4 inch. The planting depth can be set using the adjustable screw at the bottom of the seed planter - ensure this is set to an acceptable depth throughout planting as it can shift. Also, make sure that the chain is not tangled at the bottom of the planter, as it is meant to cover the soil after the planter penetrates the soil during planting. If the chain is tangled, seeds will not be covered with soil, and thus, may not germinate or be blown/washed away. Prior to using the planters, run a generous amount of graphite powder through the funnel and tube to reduce static that might cause seeds to stick to the tube walls.

The Earthway planters were modified by fitting them with funnels and tubing that allows the user to manually hand-feed the small lettuce seeds instead of using the provided seed container and plates. Planting is carried out by members of the Pauli, Arnold, and Michelmore labs. People are paired up with one person responsible for planting the seeds with the Earthway planter, and the other responsible for ensuring the correct plot numbers are being planted and that the correct seed is provided to the person planting ([Figure 3 Right](#)).



Figure 1: South gantry field with shaped raised beds, sprinkler irrigation, and strings and stakes.

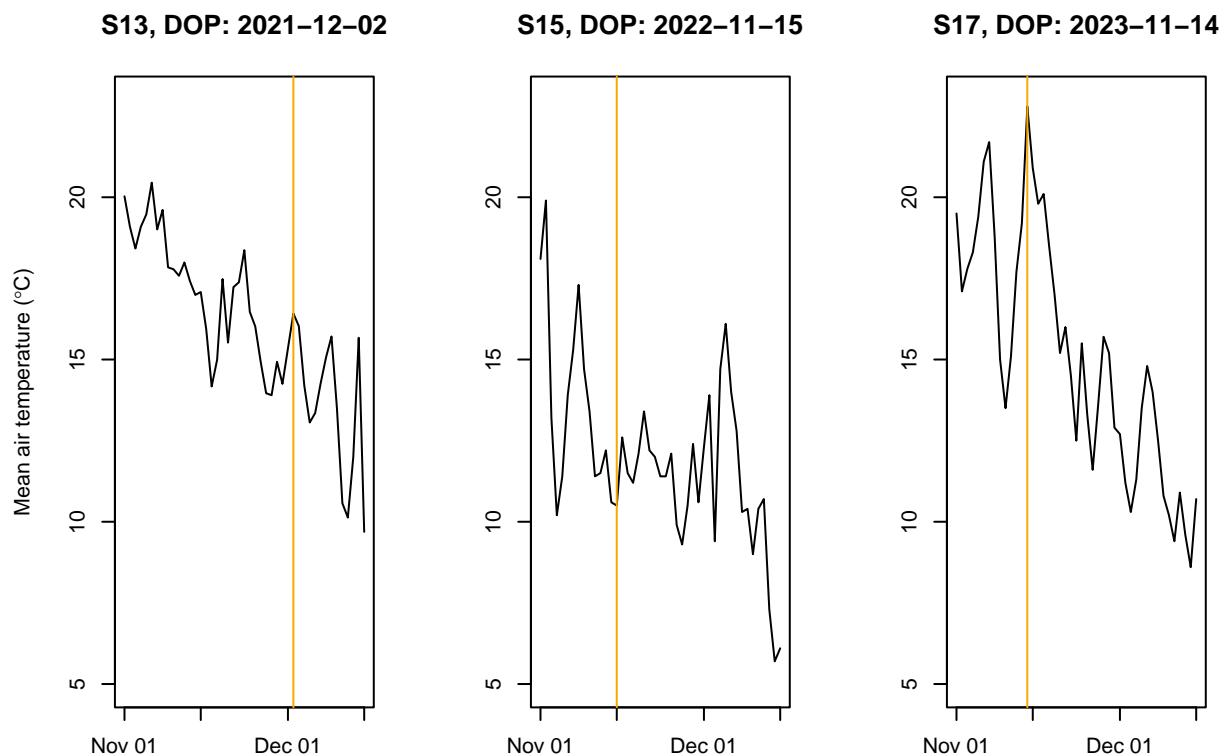


Figure 2: Mean air temperature data collected by The Arizona Meteorological Network (AZMET). Orange vertical lines represent the day of planting. DOP, day of planting; S13, season 13; S15, season 15; S17, season 17.



Figure 3: Lettuce hand planting. (Left) Earthway garden seed planter. (Right) One person planting using the Earthway planter, while the other is responsible for ensuring correct plot numbers and handing the correct seed to the person planting.

1.1.2 Potential Issues During Planting

In past years, the tubing that feeds the seeds into the ground have gotten pinched or otherwise clogged. In these cases, entire columns were inadequately planted - the seed did not make it into the seed line of the expected plot. When this happens, Drs. Duke Pauli and Maria José Truco are notified. The plots within the specific column/s are noted. If seed is not immediately available, Dr. Maria José Truco sends it from Davis, California.

1.2 Ground Control Points

The raw data collected by the Field Scanalyzer has a high level of misalignment of images and point clouds. To mitigate this error, a high number of ground control points (GCPs) are placed in the field. These GCPs include (**Figure 4**): - White plastic bucket lids, four columns into the field on both east and west ends. Located in the alleys of the plot without puncturing the drip tape. - Umbrella holders with grey metal bucket lids, furrow between four and five columns into the field on both east and west ends

Each range contains a single white plastic bucket lid and two umbrella holders with grey metal bucket lids in the following arrangement (**Figure 5**):



Figure 4: Ground control points (GCPs) used in the gantry field. (Left) White plastic bucket lid. (Right) Umbrella holder with grey metal bucket lid.

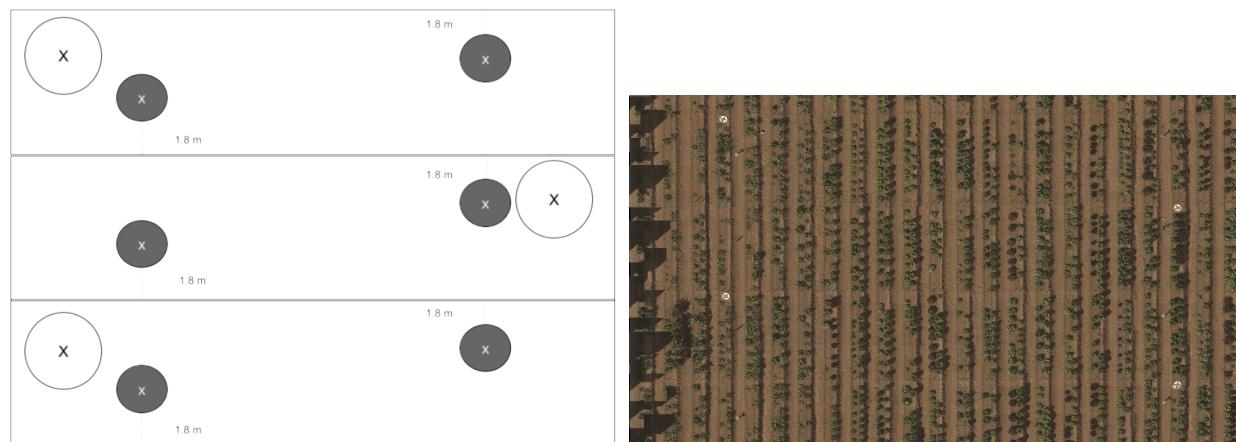


Figure 5: Arrangement of ground control points (GCPs) in the gantry field. (Left) Each range contains a single white plastic bucket lid and two umbrella holders with grey metal bucket lids. (Right) White plastic bucket lids are alternated, to ensure robust geocorrection.

1.3 Thinning

Thinning is a very important part of the field trial as the planters often result in clusters of seeds germinating close to each other. Thinning is conducted in two phases (**Figure 6**):

- Phase 1: Thin the plots to achieve 6-inch spacing between plants (results in ~20 plants per plot)
- Phase 2: Thin the plots to achieve 12-inch spacing between the plants (results in ~10 plants per plot), with the goal of staggering plants between plots

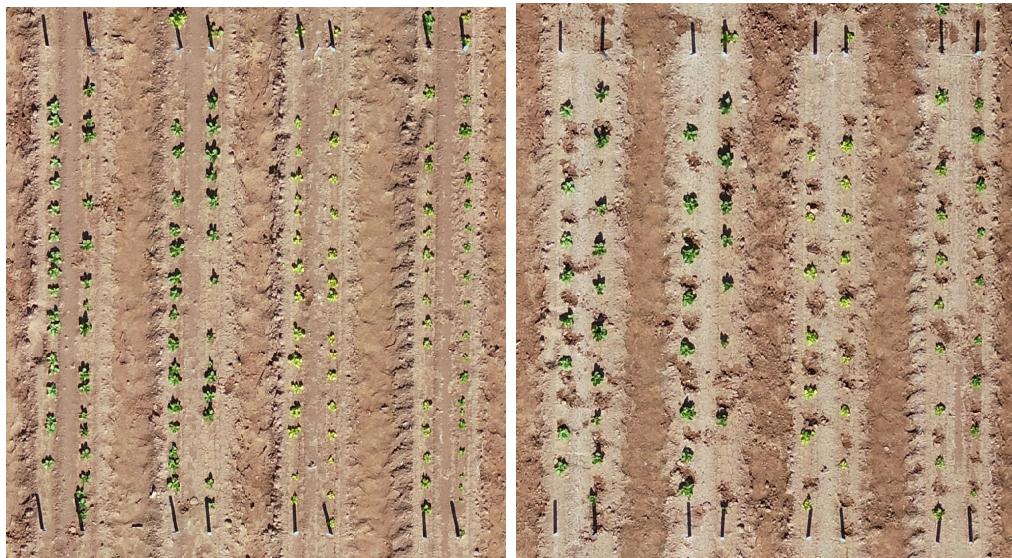


Figure 6: Change in plant density after multiple rounds of thinning. (Left) Plants after Phase 1 of thinning. (Right) Plants after Phase 2 of thinning.

The 10 individual plants resulting from Phase 2 should be equidistant. The equidistant placement reduces any overlap with neighboring plants. This is an important step as the goal with the Field Scanalyzer data is to phenotype each plant individually. The farther plants are, the easier it is to individually phenotype them.

2 Positioning Information Preparation

The Global Positioning System (GPS) coordinates of each GCP must be collected so they can be used in PhytoOracle workflows. To accomplish this, you need a Trimble Global Navigation Satellite System (GNSS) (**Figure 7**).



Figure 7: Trimble Global Navigation Satellite System (GNSS) used to collect accurate Global Positioning System (GPS) coordinates of Ground Control Points (GCPs).

2.1 Collecting Global Positioning System (GPS) Coordinates

The United States Department of Agriculture (USDA) Arid Land Agricultural Research Center (ALARC) has trimbles that we can borrow. To use them, follow the steps below:

1. Run Trimble Access - Press Trimble hard key (Windows symbol), select Trimble Access
 2. Log in — Click either “Tap here to log in” or the current logged in person (e.g., kelly.thorp)
- Login Mode: Offline

- Select your user name.
 - If you already have a username, select it...click next...click finish
 - Else type in your user name...Click finish
 - Passwords are optional...be professional

3. Set up a job - Click General Survey -> Jobs

- If opening existing job, click “Open job” then select the job
- If starting a new job, click “New job” then
 - Job name: Give the job a name.
 - Template: Default
 - Coord sys.: Click ‘Select from library’
 - * System: ‘World wide/UTM’
 - * Zone: 12 North
 - * Datum: WGS 1984 (7P)
 - * Project height: 361m
 - Click Store Accept
- If using the Yuma 2 and the Bluetooth needs connected
 - Turn on the GNSS receiver
 - General Survey -> Instrument -> GNSS Functions Bluetooth
 - * Under ‘Connect to GNSS Rover’ select: R8s, Serial#: Trimble
 - * Save/Accept

4. To measure points

- Go to field. Click General Survey -> Measure -> ALARCRTK -> Measure points
- Accept base station.
- Point name: Name the point. (If you put a number at the end, it will auto-increment.)
- Code: Add optional additional info.
- Method: Rapid point
- Antenna height (Uncorr): 2.000m
- Vertical offset: 30.839m Up (This is important to get the Z coordinate correct.)
- Click “Measure” to record a point.
- Go to lab. Connect TSC3 to computer.
- Click General Survey -> Jobs -> Import/Export -> Export fixed format -> Accept -> All points
- Copy the file off the TSC3:
 - “This PC\Trimble Navigation Limited TSC3\Export”

5. To stake flags at point locations

- Copy CSV file to the TSC3
 - Save CSV from Excel to help TSC3 recognize points. Do not add header line.
 - Order: Point name, Northing, Easting, Elevation, Code (optional)
 - Connect TSC3 to computer and copy point file to your user directory.
 - * “This PC\Trimble Navigation Limited TSC3\”
- Go to field. Click General Survey -> Stakeout -> ALARCRTK -> Points - Remove points from old stakeouts, if they exist. - Add -> Select from file -> choose point file. - Click ‘All’ - Click Add - Select a point, then select StakeoUt Navigate to the point, and stake flag.

2.2 Positioning Information Files Required by PhytoOracle

PhytoOracle relies on geospatial information, such as GPS coordinates, to accurately link phenotypes with a location in the field. This allows us to detect, tag, and track individual plants over the course of multiple Field Scanalyzer scans. Specifically, PhytoOracle requires two files:

1. GCP file: Text file containing the GPS coordinates of all field GCPs.
2. GeoJSON: File containing polygons representing each plot in the gantry field

These files must be generated prior to data processing for the respective season. Additionally, these files should be loaded onto [QGIS](#) for visual inspection and confirmation that the coordinates are accurate.

2.2.1 Generating GCP File

The Trimble collects GPS coordinates in the Easting, Northing format (Table 1). PhytoOracle requires GPS coordinates to be in the latitude, longitude format. To convert the coordinates, use the [gcp_coordinates_conversion repository](#) to use the conversion tool. After running the conversion script, the data will now be in the required latitude, longitude format (Table 2).

2.2.2 Generating GeoJSON File

GeoJSON files contain polygons that represent each plot in the gantry field ([Figure 8](#)). These polygons are used to extract smaller experimental units from larger units, such as the full field scale.

Our field design and dimensions remain pretty consistent from one season to the next. As a result, existing GeoJSONs are modified and applied to new seasons. In the case that a new GeoJSON needs to be created, please refer to [FIELDimageR](#).

If you are editing a pre-existing GeoJSON, you will need to:

Table 1: Ground Control Point (GCP) coordinate file. Each row represents the coordinate of a single GCP.

GCP	Type	Northing	Easting	Height..m.
plate1	White	3659979	408992.8	360.775
plate2	White	3659987	408992.9	360.788
plate3	White	3659995	408992.9	360.783
plate4	White	3660003	408993.0	360.770
plate5	White	3660011	408993.1	360.775
plate6	White	3660019	408993.1	360.765

Table 2: Ground Control Point (GCP) coordinate file. Each row represents the coordinate of a single GCP.

GCP number	Latitude	Longitude
1	33.07470	-111.975
2	33.07478	-111.975
3	33.07485	-111.975
4	33.07492	-111.975
5	33.07499	-111.975
6	33.07506	-111.975

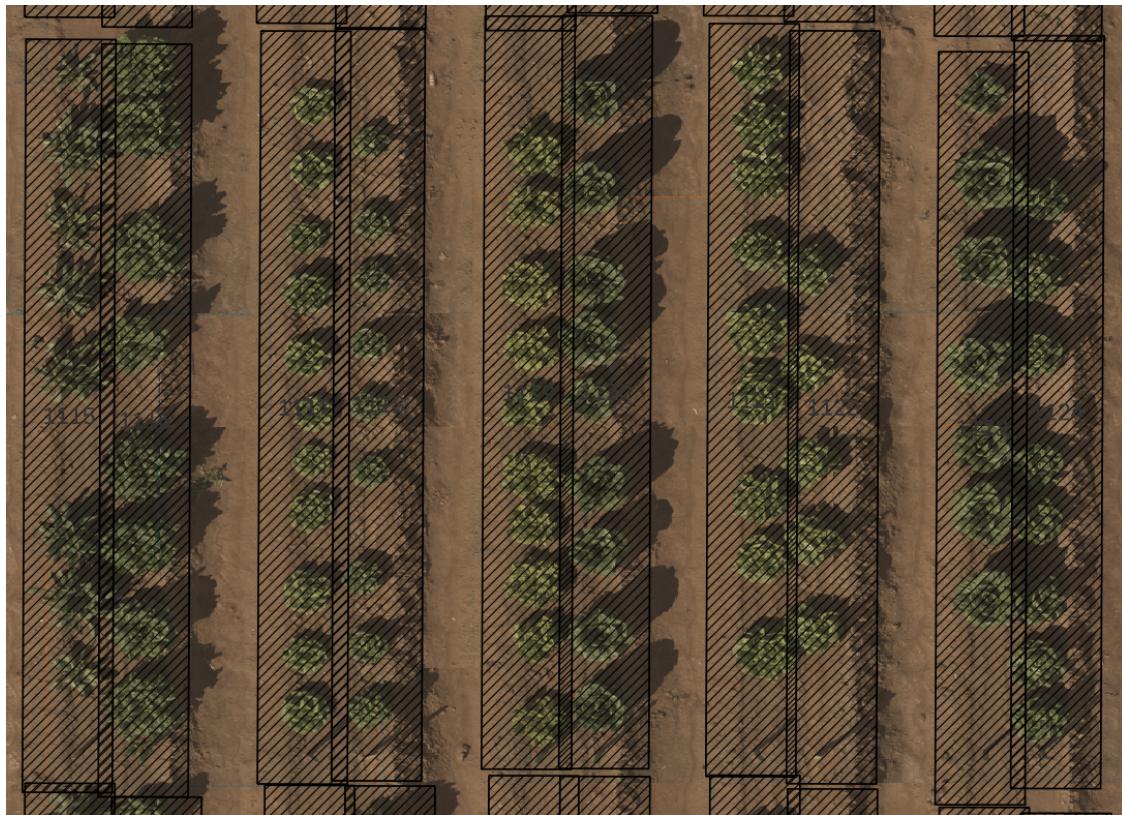


Figure 8: GeoJSON file containing a single polygon for each plot.

1. Move polygons that are misaligned in the new season
2. Rename genotype column

2.2.2.1 Moving polygons To move polygons, you need to load the GeoJSON and a drone orthomosaic onto QGIS. Then, you can follow the steps in **Figure 9**:

1. Click “Toggle Editing”
2. Click “Select Features by Area or Single Click”
3. Click “Move Features”
4. Manually move polygon into desired alignment
5. Single click to drop the polygon into the desired location
6. Save changes

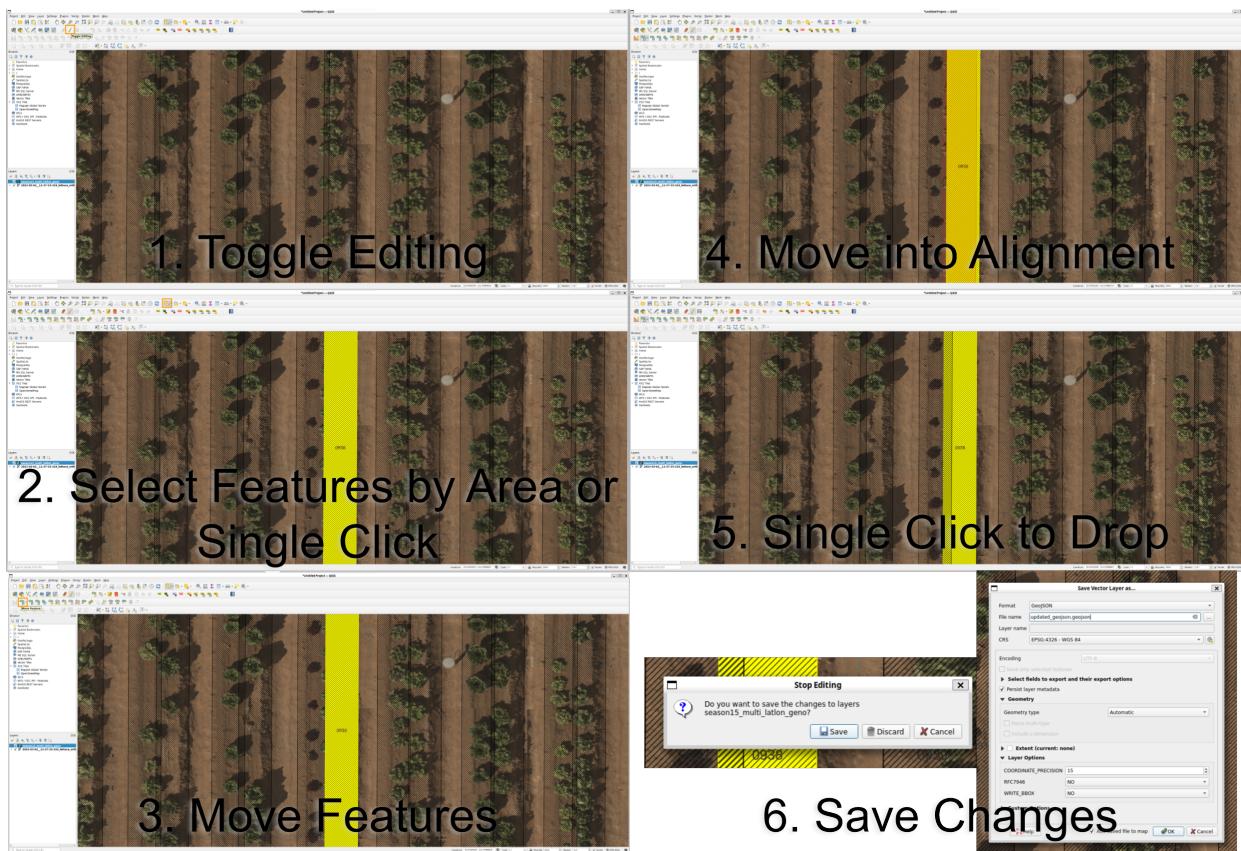


Figure 9: Editing GeoJSON polygons using QGIS.

2.2.2.2 Renaming genotype column The “genotype” values in the GeoJSON file can be edited using [GeoPandas](#). A GeoJSON can be opened up as a dataframe, similar to [Pandas](#). Once opened, you can then replace the “genotype” columns using the fieldbook for the respective season. To see an example [click here](#).

3 Pipeline Preparation

The PhytoOracle (PO) pipelines require the aforementioned GCP and GeoJSON files. Additionally, a Yet Another Markup Language (YAML) file is used by PO for automated, reproducible data processing. YAMLs are a form of a configuration file that can be used to define a series of arguments/flags. The details of the YAML files can be found on our [PhytoOracle Automation repository](#).

3.1 Editing YAML file

For each season, YAML files must be edited to correctly process data for the respective season.

Specifically, the following keys should be edited for each season:

- tags
 - pipeline
 - season
 - season_name
- workload_manager
 - manager_name
- paths
 - cyverse
 - * input
 - necessary_files

Examples of YAMLs for each season can be found [here](#).

3.2 Updating GitHub Repositories

3.2.1 PhytoOracle Data

At the start of a new season, updates must be made to the phytooracle_automation repository to capture necessary information in the [season_config_yaml variable](#).

The season is defined by multiple keys, including name, start_date, end_date, flir_temp_units, and complete_field_dates ([Figure 10](#)).

Below are some details for each key:

- name: Season name, as it shows up on the CyVerse DataStore

- start_date: Date of planting
- end_date: Date of harvest
- flir_temp_units: Temperature units reported by thermal (FLIR) images
- complete_field_dates: List of RGB scan dates

To get a list of RGB dates, use iRODS to ils the directory of the respective season (**Figure 11**).

3.2.2 PhytoOracle Landmark Selection

The PhytoOracle [3d_landmark_selection](#) contains the phytooracle_data repository. As such, the [3d_landmark_selection container on DockerHub](#) must be rebuilt once the above-mentioned changes have been made to the phytooracle_data repository. To accomplish this, log into an account with appropriate permissions, then click on “Trigger” for the “latest” container (**Figure 12**).

```
181    13:  
182        name: season_13_lettuce_yr_2022  
183        start_date: 2022-11-14  
184        end_date: 2023-03-30  
185        flir_temp_units: "C"  
186        complete_field_dates:  
187            rgb:  
188                ['2022-01-27_10-54-27-164_lettuce',  
189                    '2022-01-31_10-45-39-463_lettuce',  
190                    '2022-02-01_10-39-38-575_lettuce',  
191                    '2022-02-04_10-42-09-085_lettuce',  
192                    '2022-02-07_10-33-28-355_lettuce',  
193                    '2022-02-08_10-38-14-413_lettuce',  
194                    '2022-02-09_10-42-41-102_lettuce',  
195                    '2022-02-10_10-37-53-213_lettuce',  
196                    '2022-02-11_10-37-09-800_lettuce',  
197                    '2022-02-14_10-41-53-763_lettuce',  
198                    '2022-02-16_10-43-57-766_lettuce',  
199                    '2022-02-17_10-39-08-670_lettuce',  
200                    '2022-02-18_10-41-21-294_lettuce',  
201                    '2022-02-21_10-38-28-308_lettuce',  
202                    '2022-02-25_10-39-57-210_lettuce',  
203                    '2022-02-28_10-38-14-633_lettuce',  
204                    '2022-03-02_10-56-11-405_lettuce',  
205                    '2022-03-07_10-38-37-481_lettuce',  
206                    '2022-03-09_10-40-08-384_lettuce',  
207                    '2022-03-11_10-39-22-200_lettuce',  
208                    '2022-03-15_11-12-27-310_lettuce']  
209
```

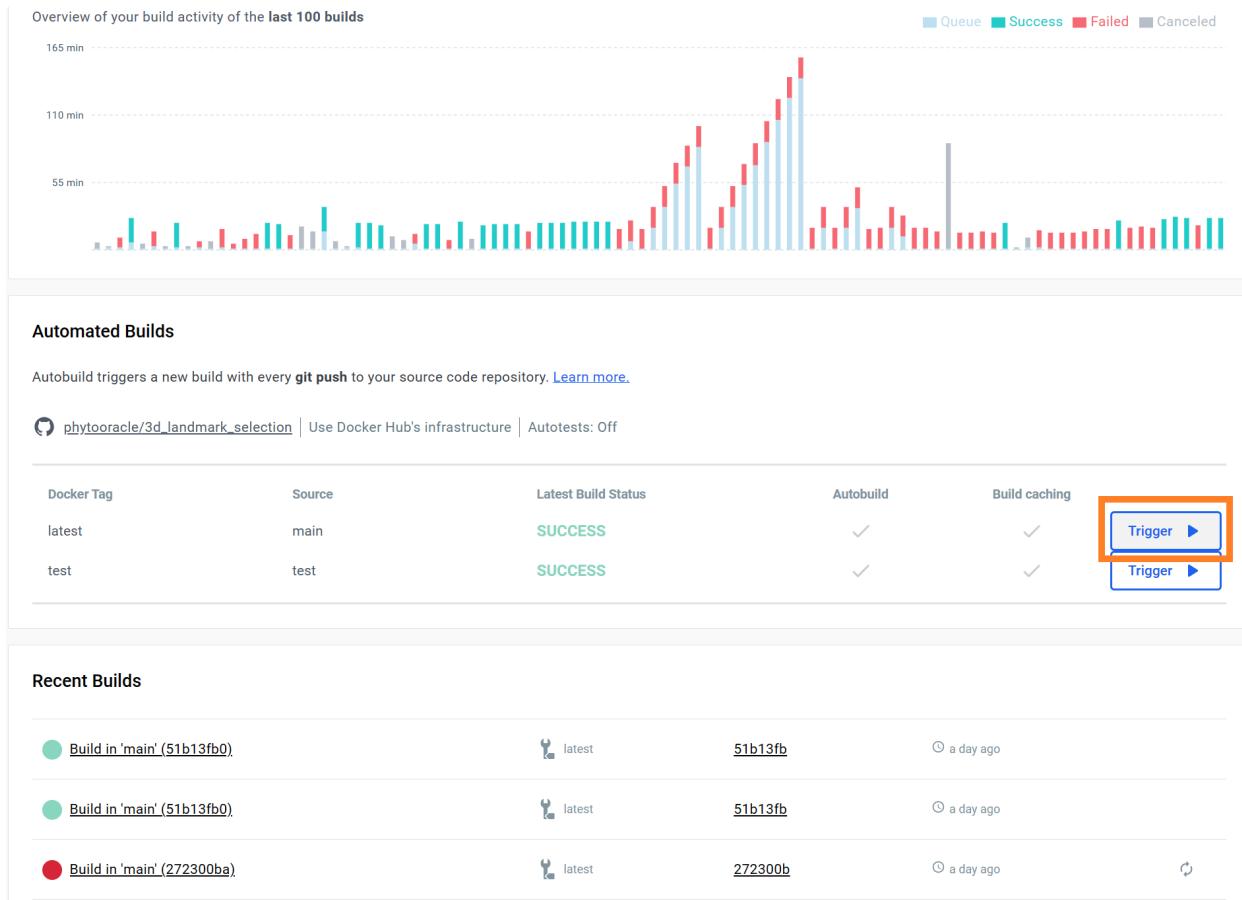
Figure 10: Section of the `season_config.yaml` variable in the `phytooracle_data` GitHub repository.

3.2 Updating GitHub Repositories



```
eg@myosotis: ... /ubuntu_files $ ils /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/ | grep 2022
iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce:
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-01-27_10-54-27-164_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-01-31_10-45-39-463_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-01_10-39-38-575_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-04_10-42-09-085_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-07_10-33-28-355_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-08_10-38-14-413_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-09_10-42-41-102_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-10_10-37-53-213_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-11_10-37-09-800_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-14_10-41-53-763_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-16_10-43-57-766_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-17_10-39-08-670_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-18_10-41-21-294_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-21_10-38-28-308_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-25_10-39-57-210_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-28_10-38-14-633_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-03-02_10-56-11-405_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-03-07_10-38-37-481_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-03-09_10-40-08-384_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-03-11_10-39-22-200_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-03-15_11-12-27-310_lettuce
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-01-01_11-24-18-691_sorghum
C- /iplant/home/shared/phytooracle/season_13_lettuce_yr_2022/level_1/stereoTop/lettuce/2022-02-02_00-00-00-000_lettuce
```

Figure 11: Getting RGB dates using iRODS.



Overview of your build activity of the last 100 builds

Queue Success Failed Canceled

165 min

110 min

55 min

Automated Builds

Autobuild triggers a new build with every `git push` to your source code repository. [Learn more.](#)

phytooracle/3d_landmark_selection | Use Docker Hub's infrastructure | Autotests: Off

Docker Tag	Source	Latest Build Status	Autobuild	Build caching
latest	main	SUCCESS	✓	✓
test	test	SUCCESS	✓	✓

Trigger ➔

Trigger ➔

Recent Builds

Build	Tag	ID	Time Ago
Build in 'main' (51b13fb0)	latest	51b13fb	a day ago
Build in 'main' (51b13fb0)	latest	51b13fb	a day ago
Build in 'main' (272300ba)	latest	272300b	a day ago

Figure 12: Rebuilding the `3d_landmark_selection` container on DockerHub.

4 Intro to High Performance Computers

4.1 UArizona High Performance Computing Cluster

The University of Arizona maintains a High Performance Computing (HPC) center, which houses three compute resources: El Gato, Ocelote, and Puma.

4.1.1 Compute System

Name	El Gato	Ocelote	Puma
<i>Model</i>	IBM System X iDataPlex dx360 M4	Lenovo NeXtScale nx360 M5	Penguin Altus XE2242
<i>Node Count</i>	131	400	236 CPU-only, 8 GPU, 2 high-memory
<i>Total System Memory (TB)</i>	26TB	82.6TB	128TB
<i>Cores / Node (schedulable)</i>	16c	28c (48c - High-memory node)	94c
<i>Total Cores</i>	2160*	11528*	23616*
<i>Processor Speed</i>	2.66GHz	2.3GHz (2.4GHz - Broadwell CPUs)	2.4GHz
<i>Memory / Node</i>	256GB - GPU nodes, 64GB - CPU-only nodes	192GB (2TB - High-memory node)	512GB (3TB - High-memory nodes)
<i>GPU accelerators</i>		46 NVIDIA P100 (16GB)	29 NVIDIA V100S
<i>OS</i>	Centos 7	CentOS 7	CentOS 7

4.1.2 Compute Resources

The UArizona HPC provides three types of resources:

- Windfall: Unlimited, can be preempted
- Standard: Limited, no preemption
 - El Gato: 7,000 CPU-hours per month
 - Ocelote: 70,000 CPU-hours per month
 - Puma: 100,000 CPU-hours per month
- High Priority:
 - Puma

- * ericlyons: 175,200 CPU-hours per month
- * dukepauli: 35,040 CPU-hours per month

*Note: High priority is only available for the Puma cluster.

4.1.3 Further Information

Further information on the UArizona HPC is available in the [HPC Documentation](#).

5 Running PhytoOracle on High Performance Computers

PhytoOracle is a scalable, modular phenomics data processing workflow manager. In short, this means that PhytoOracle can leverage HPC clusters and cloud computing to distribute tasks across hundreds to thousands of cores.

5.1 Defining Compute Resources

Resources are defined in the `workload_manager` section of the PhytoOracle YAML. In this section, you can define many compute resource settings. Below is an example:

- account: ericlyons
- high_priority_settings:
 - use: True
 - qos_group: user_qos_ericlyons
 - partition: high_priority
- standard_settings:
 - use: False
 - partition: standard
- job_name: phytooracle_worker_rgb
- nodes: 1
- number_worker_array: 490
- cores_per_worker: 1
- time_minutes: 720
- retries: 1
- port: 0
- mem_per_core: 5
- manager_name: stereoTop_level01_s15
- worker_timeout_seconds: 43200

5.2 Before Deploying PhytoOracle

There are a few things you must ensure before deploying PhytoOracle:

- Confirm existence and accuracy of GCP file. Visually inspect using GIS software (e.g., QGIS), confirming correct placement of GCPs by overlaying the points with an RGB orthomosaic, either drone or gantry.
- Confirm existence and accuracy of GeoJSON file. Visually inspect using GIS software (e.g., QGIS), checking plot number sequence and genotype values.

If these steps are not followed, errors can propagate to multiple levels of data processing, requiring a reprocessing of data.

5.3 Supported Data Types

The Field Scanner collects two-dimensional (2D) and three-dimensional (3D) data types, including scannerTop3D (3D), stereoTop (RGB), ps2Top (fluorescence), and flirIrCamera (thermal) ([Figure 13](#)).

5.3.1 2D Field Scanner Data Types

The 2D data collected by the Field Scanner includes stereoTop (RGB), flirIrCamera (thermal), and ps2Top (fluorescence). These data process relatively quickly as they are much lower in size compared to 3-dimensional (3D) data. The processing of 2D data types is fully developed for both lettuce and sorghum ([Figure 14](#)).

5.3.2 3D Field Scanner Data Types

The major goal of the PhytoOracle project is to phenotype individual plants at a high spatial-temporal scale. To accomplish this, individual plant positioning information (GPS coordinates) collected during 2D data processing are leveraged to extract data from 3D data ([Figure 15](#)).

As such, much focus has been placed on 3D point cloud data. These data undergo intensive processing to extract individual plant point clouds ([Figure 16](#)).

5.4 Deploying PhytoOracle

After (*i*) checking the GCP and GeoJSON files (Section [5.2](#)) and (*ii*) generating a YAML file (Section [3.1](#)), you are now ready to run PhytoOracle.

PhytoOracle is made up of multiple workflows to process 2D and 3D data ([Figure 17](#)). These workflows allow for automated, scalable processing of raw data collected by the Field Scanner. The data processing results in high spatial-temporal phenotype information.

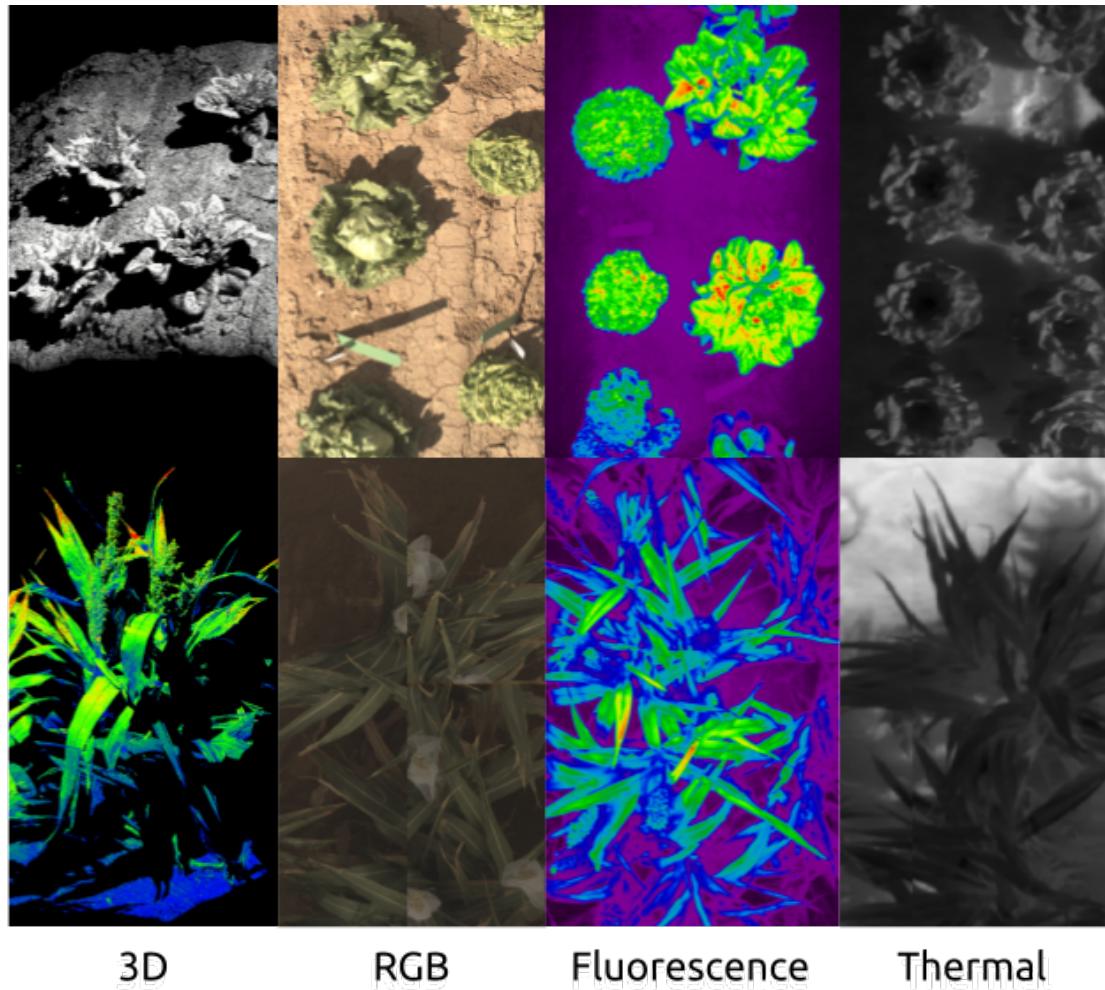


Figure 13: Data types collected by the Field Scanner. Two-dimensional (2D) data types include RGB, fluorescence, and thermal images, while three-dimensional (3D) include 3D point clouds.

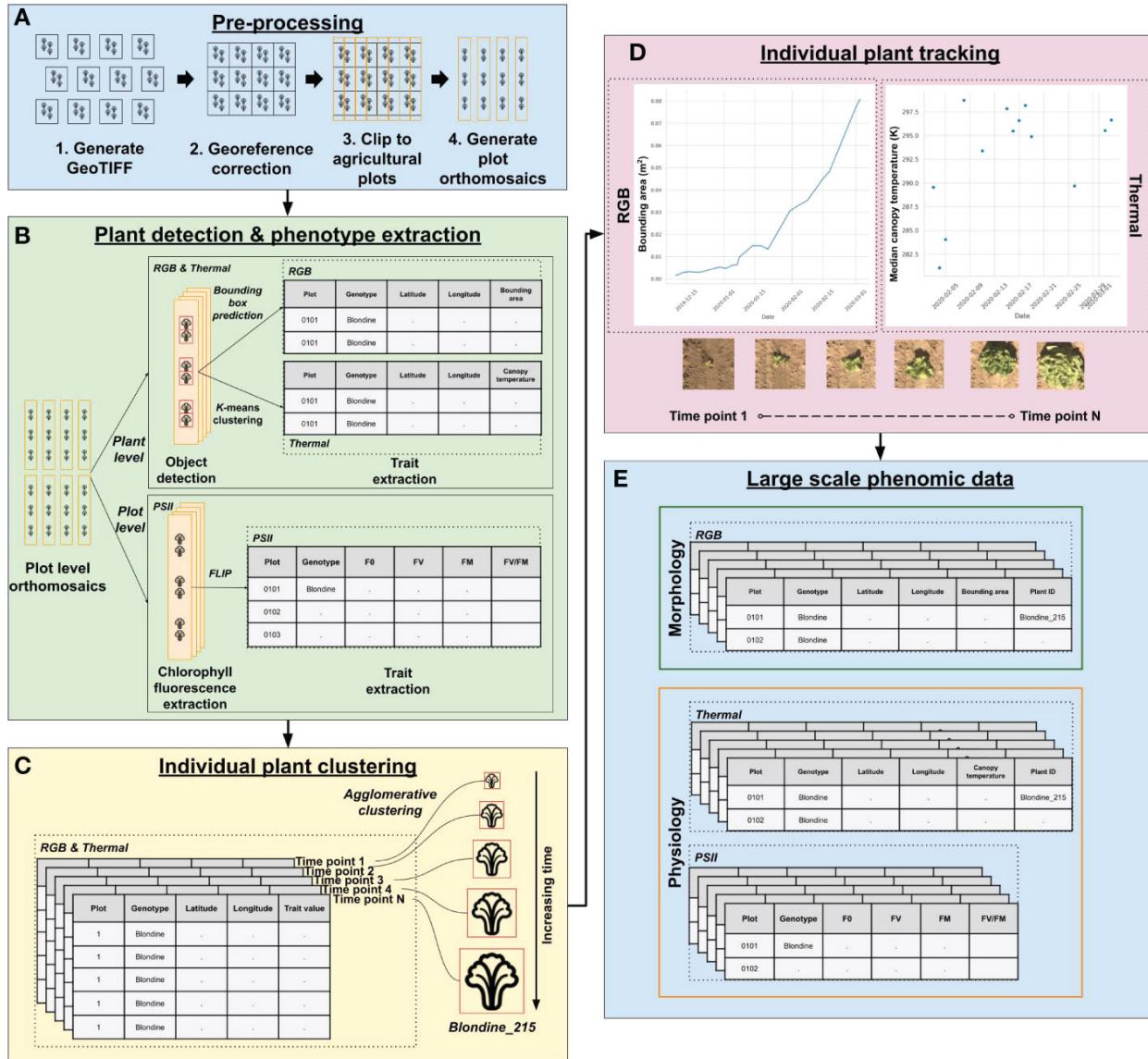


Figure 14: Visualization of 2D data processing by PhytoOracle.

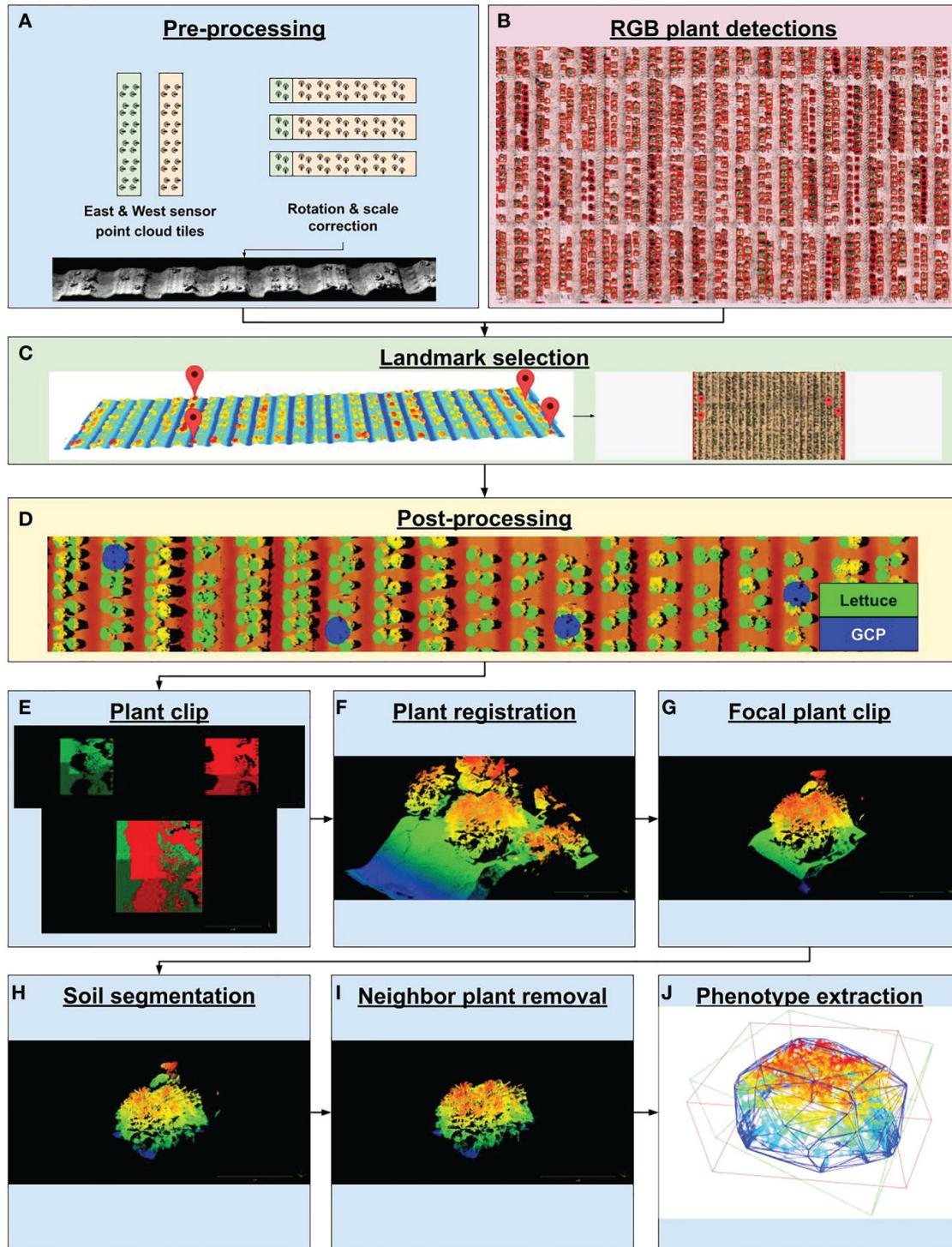


Figure 15: Visualization of 3D data processing by PhytoOracle.

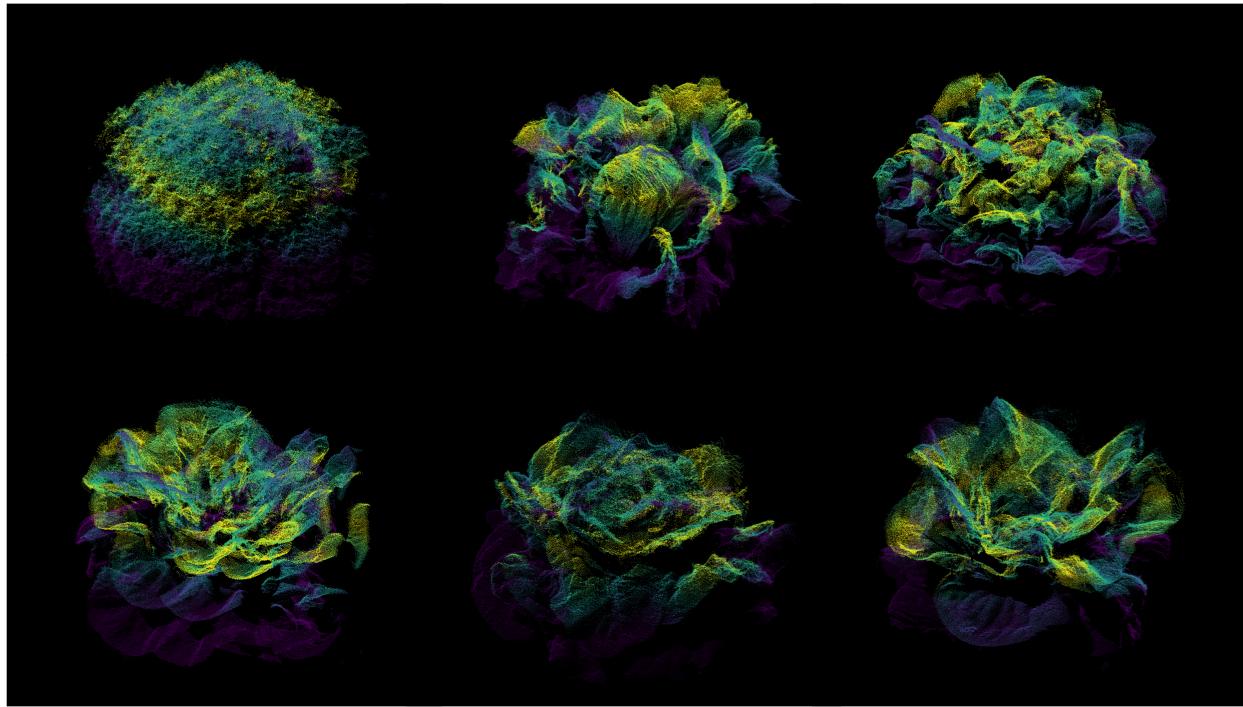


Figure 16: Individual plant point clouds processed by PhytoOracle.

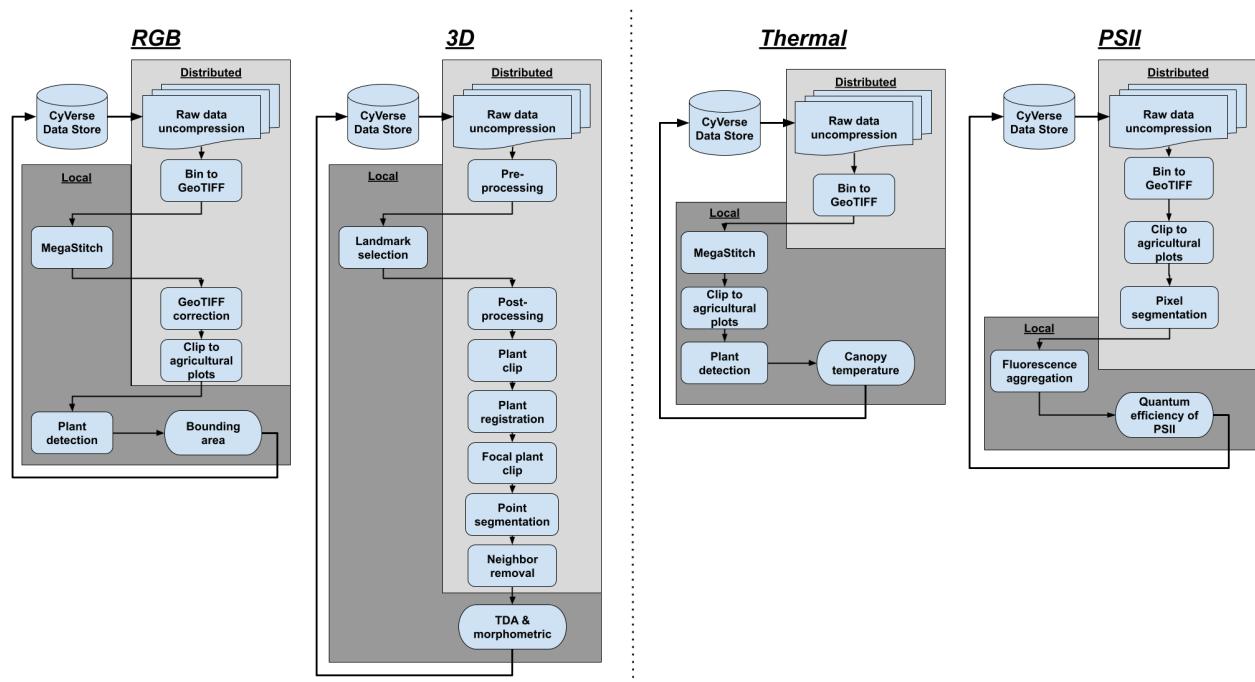


Figure 17: PhytoOracle workflows for processing raw data collected by the Field Scanner.

PhytoOracle is mainly deployed on the UArizona HPC. The next sections provides a brief description of how to run each workflow. For additional details, please refer to the [PhytoOracle publication](#). In all cases, the commands provided will automatically handle all steps of processing, including:

- Raw data download
- Raw data extraction (.tar file)
- Launching workers
- Output archiving (.tar file)
- Output upload onto [CyVerse DataStore](#)

Prior to running any pipelines, users must complete the following:

- Set up a terminal client
- Register for an HPC account
- Create a CyVerse account
- Configure SSH keys

Prior to running any pipelines, the project lead must complete the following:

- Request access for the user to PI /xdisk/
- Grant permissions to directories on CyVerse

5.4.1 stereoTop

The stereoTop workflow runs image stitching and plant detection, resulting in the extraction of bounding area and GPS coordinate information for each plant. The workflow is run as follows:

```
sbatch shell_scripts/slurm_submission_large.sh <yaml_file>
```

For example, if you wanted to run this for season 15:

```
sbatch shell_scripts/slurm_submission_large.sh  
yaml_files/season_15/stereoTop_level01_s15.yaml
```

5.4.2 flirIrCamera

The flirIrCamera workflow runs image stitching and plant detection, resulting in the extraction of canopy temperature and GPS coordinate information for each plant. The workflow is run as follows:

```
sbatch shell_scripts/slurm_submission_large.sh <yaml_file>
```

For example, if you wanted to run this for season 15:

```
sbatch shell_scripts/slurm_submission_large.sh  
yaml_files/season_15/flirIrCamera_level01_s15.yaml
```

5.4.3 ps2Top

The ps2Top workflow applies a threshold to fluorescence plot-centered images, resulting in the extraction of maximum potential quantum efficiency of Photosystem II (Fv/Fm).

```
sbatch shell_scripts/slurm_submission_large.sh <yaml_file>
```

For example, if you wanted to run this for season 15:

```
sbatch shell_scripts/slurm_submission_large.sh  
yaml_files/season_15/ps2Top_level01_s15.yaml
```

5.4.4 scanner3DTOP

The scanner3DTOP workflow runs point cloud stitching leverages GPS coordinates collected during stereoTop processing, resulting in the extraction of traditional and topological shape descriptors for each plant. This workflow involves multiple levels of processing, including:

- Generation of test datasets using the [test_dataset_automator repository](#)
- Transformation selection using the [3d_transformation_selection repository](#)
- Level 01 processing on the UArizona HPC
- Landmark selection using the [3d_landmark_selection repository](#)
- Level 02 processing on the UArizona HPC

For example, if you wanted to run level 1 processing for season 15:

```
sbatch shell_scripts/slurm_submission.sh  
yaml_files/season_15/scanner3DTOP_level01_s15.yaml
```

To run level 2 processing for season 15:

```
sbatch shell_scripts/slurm_submission.sh  
yaml_files/season_15/scanner3DTOP_level02_s15.yaml
```

*Note: Notice that scanner3DTop level 1 and 2 processing uses the shell_scripts/slurm_submission.sh instead of shell_scripts/slurm_submission_large.sh. This is because the manager node performs no processing, it merely provides the tasks and sends them to worker nodes. As such, the manager node only requires two cores instead of 94.

5.5 Quality Control & Quality Assurance of Processed Data

Although PhytoOracle is reproducible due to the use of containers and YAML configuration files, it is important to follow quality control (QC) and quality assurance (QA) steps after data processing. The recommended steps for this are:

- Check Slack notifications (currently Cyverse/gantry_data_updates) to check for reported errors
- Check the [CyVerse DataStore](#) to confirm upload
- Visualize stereoTop, flirIrCamera orthomosaics (ending in *.tif) on [QGIS](#)
- Pull level 2 scanner3DTop (individual plant point clouds) for stretching/bending or neighboring plant material (*_segmentation_pointclouds.tar)

If any errors are spotted during these QA/QC steps, immediately notify the project lead. Depending on the impact of the error, data may need to be reprocessed to ensure data integrity.

6 Data Visualization

6.1 PhytoOracle Dashboard

PhytoOracle results in vast amounts of phenotypic information, which can be used to study plant growth patterns, responses to biotic and abiotic stress, and identification of top-performing genotypes. To enable access to these information, the PhytoOracle team has developed a [Streamlit](#) app which can be accessed [here](#).

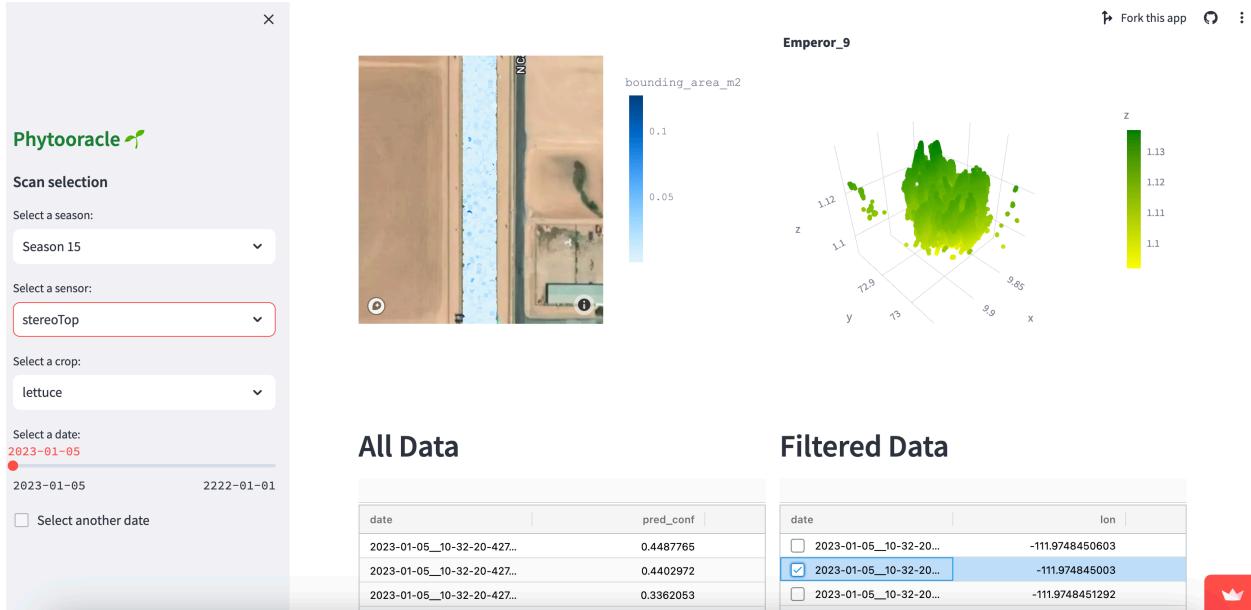


Figure 18: PhytoOracle dashboard app on Streamlit.

The dashboard is under development at the [streamlit_dashboard](#) GitHub repository, and changes are expected. Currently, the dashboard allows users to access, filter, and download phenotype information (**Figure 18**). This dashboard is useful to: (i) check quality and integrity of pipeline outputs, (ii) access data for research purposes, (iii) share progress and phenotypic information with stakeholders.