

Operations & phenotyping //

## Manuals

# Standard operating procedures for UAV phenotyping

- ▶ This manual provides an introduction and use guide to the aerial platforms, sensors and data tools involved in unmanned aerial vehicle (UAV) phenotyping applications, with a case study of RGB, multispectral and thermal phenotyping on microplots.
- ▶ Published on 01/12/2019

[Excellenceinbreeding.org/toolbox/tools/standard-operating-procedures-uav-phenotyping](https://Excellenceinbreeding.org/toolbox/tools/standard-operating-procedures-uav-phenotyping)



**Excellence  
in Breeding**  
PLATFORM

[ExcellenceinBreeding.org](https://ExcellenceinBreeding.org)



# Basic standard operating procedures for UAV phenotyping

## Authors and editors

**Dr. Shawn Carlisle Kefauver**, Researcher / [sckefauver@ub.edu](mailto:sckefauver@ub.edu)

**Prof. José Luis Araus Ortega**, Full Professor / [jaraus@ub.edu](mailto:jaraus@ub.edu)

**Ms. M. Luisa Buchailot**, PhD Student

University of Barcelona, Faculty of Biology, Plant Physiology Section

Integrative Crop Ecophysiology Group, Av. Diagonal 643. 08028 Barcelona, Spain

**Vincent Vadez**, Principal Scientist (Directeur de Recherche) / [vincent.vadez@ird.fr](mailto:vincent.vadez@ird.fr)

IRD (Institut de Recherche pour le Développement) – UMR DIADE – Team CERES

911 Av Agropolis – BP 64501 – 34394 Montpellier Cedex 5, France

## Conceptual guidance provided by:

**Federic Baret**, INRA Avignon, France / [frederic.baret@inra.fr](mailto:frederic.baret@inra.fr)

**Mainassara A. Zaman-Allah**, CIMMYT South Africa Regional Office, Harare, Zimbabwe /

[Z.MainassaraAbdou@cgiar.org](mailto:Z.MainassaraAbdou@cgiar.org)

**Jill Cairns**, CIMMYT South Africa Regional Office, Harare, Zimbabwe / [J.Cairns@cgiar.org](mailto:J.Cairns@cgiar.org)

**Steven Klassen**, IRRI/CGIAR / [s.klassen@irri.org](mailto:s.klassen@irri.org)

**Scott Chapman**, CSIRO and The University of Queensland / [scott.chapman@csiro.au](mailto:scott.chapman@csiro.au)

**Andries Potgieter**, The University of Queensland / [a.potgieter@uq.edu.au](mailto:a.potgieter@uq.edu.au)

**Jesse Poland**, Kansas State University, USA / [jpoland@k-state.edu](mailto:jpoland@k-state.edu)



Supported in part by the University of Montpellier MUSE International Mobility Program.

# Contents

<b>Before you buy: A brief introduction to UAV phenotyping systems .....</b>	<b>1</b>
A. Evaluate readiness level .....	1
B. Sensors .....	1
C. Unmanned Aerial Vehicle (UAV) Platforms.....	4
D. Software .....	5
<b>Data capture .....</b>	<b>6</b>
A. Basics before going to the field.....	7
B. Airborne data capture during a UAV flight .....	8
<b>Data processing.....</b>	<b>20</b>
A. Post-UAV flight data check.....	20
B. Data quality check by sensor type .....	20
C. Orthomosaic generation and georeferencing using the structure from motion pipeline ....	21
D. Image Radiometric Calibration .....	28
<b>Data extraction.....</b>	<b>31</b>
A. Trimming the orthomosaic to micro-plots.....	31
B. Extraction of micro-plot information .....	32
<b>Case study: UAV phenotyping on &lt;1000 plots at the University of Barcelona .....</b>	<b>35</b>
Sensors used for image data capture .....	36
UAV image data pre-processing .....	38
MosaicTool image microplot data processing details.....	40
References .....	42
Specific code used .....	43

# Before you buy: A brief introduction to UAV phenotyping systems

## A. Evaluate readiness level

UAV phenotyping is about collecting useful and meaningful data for integration into crop breeding programs. As such, a complete UAV phenotyping program should ideally include guidelines and considerations to evaluate the readiness of every part of the full process in order to ensure that no bottlenecks impede the throughput of the phenotyping activities all the while focused on the end goal of data. This includes but is not limited to (1) the equipment (sensors, platforms and software), (2) flight operation (permits, training, flight plans, and image acquisition), (3) Proper storing and managing of experimental datasets for long term use (4) post-flight image processing (pre-processing, calibration, mosaicking), (5) data generation (extraction from processed image to plot level data), (6) data analysis (index calculations, stats scripts) and database structure (storage, linkages, inventory indexing, ontologies,...); (7) specific case studies of bottlenecks to throughput, training requirements, costs, and optimization for specific crops (scalable/transferable traits). In essence, all of the major components from sensors to platforms to software for each key processing step are intricately intertwined and need to be considered together before purchasing either one or the other. Pre-integrated systems or close adherence to integration guidelines as detailed in this set of documents will improve both data quality and data throughput.

## B. Sensors

The primary selection of equipment in the development of a high throughput plant phenotyping program for crop breeding should be the selection of the most adequate sensors for the estimation of the specific biophysical traits of interest at an appropriate technology readiness level (TRL). Thus, new UAV phenotyping operations should first consult the trait TRLs

as determined by the scientific research and development community and then select the most easily “out-scalable” ones that are capable of measuring the traits of interest to specific crop breeding programs, whether considered a complimentary additional trait or a potential win in terms of cost and efficiency. Note that this may not be the same case as seen in the scientific and research development communities, where the standard for forward looking activities of research institutions is rather testing and exploratory activities regarding what may be possible. In order to avoid bottlenecks during the integration of an aerial phenotyping component to an existing breeding program the recommended focus is on the selection of the most appropriate state-of-the-art and yet already fully understood and operational sensor technologies.

In the selection of sensor technologies there is often a trade-off between spatial and spectral resolution, a result of the specific detailed technological specifications of different types of sensors used for each type of imaging. Spatial resolution refers to the pixel size within the resulting image, which is a product of both the sensor size as well as the sensor processing to produce the resulting digital image. Spectral resolution refers to the specific wavelengths of electromagnetic radiation (in most cases this refers to sunlight, as in the case of “passive” sensors or cameras that produce digital images) that are measured. Without going in to the details of the various camera sensor technologies, the tradeoff is in most cases due to efforts to maintain an adequately high signal quality (signal to noise ratios) – with a smaller amount of incoming light/radiation to the sensor, the number of pixels that can be produced becomes smaller. An example is provided in **Table 1**, where a typical commercially available natural color RGB (red green blue) digital camera is compared with a multispectral and thermal camera.

**Table 1.** Example camera and sensor details for RGB, multispectral and thermal cameras.

Camera	Sensor size	Sensitivity	Lens	Resolution
<b>Panasonic Lumix GX7</b>	17.3 x 13.0 mm	8 bit RGB Live MOS	20 mm	4592x3448
<b>Airinov Multispec 4C</b>	4 x 6x4.8 mm	10 bit G, R, RedEdge, NIR CMOS	4 mm	1280x960
<b>TEAX TC 2.0 FLIR TAU2 640</b>	10.88x8.7 mm	Uncooled Vox Microbolometer	19 mm	640x512

It is fairly typical to see approximately 10x spatial resolution (ground spatial distance, GSD, or pixel size) of an RGB camera compared to a multispectral camera, while thermal cameras tend to have approximately half the spatial resolution of most multispectral cameras. The large ranges of radiation wavelengths (near 100 nm swaths) captured by each separate Red, Green and Blue channels of an RGB camera allow the camera to produce much higher spatial resolution images at the cost of overlapping somewhat and producing average reflectance values for a greater range reflected light.

Multispectral cameras that may measure over the same area usually offer distinct and smaller ranges of 10-20 nm provide more specific and separate measurements, but capture less incoming light and produce fewer pixels per sensor area.

Thermal cameras measure in a wholly different range of the electromagnetic spectrum where the signal is quite a bit weaker and as such need different, much more expensive sensor technologies and have to measure over a relatively wide range in order to produce consistent and high-quality images. Example electromagnetic radiation spectral sensitivity details for typical RGB, multispectral and thermal cameras are further detailed in **Table 2**.

**Table 2.** Spectral sensitivity details of the Lumix GX7 RGB natural color, the AIRINOV Multispec 4C multispectral and the FLIR Tau2 640 thermal infrared sensors. \*B=Blue, G=Green, R=Red, RE=Red Edge, NIR=Near Infrared, TIR=Thermal Infrared.

Camera	Band 1	Band 2	Band 3	Band 4
<b>Panasonic Lumix GX7</b>	B: 480 nm ± 50	G: 540 nm ± 50	R: 640 nm ±50	
<b>AIRINOV Multispec 4C</b>	G: 550 nm ± 10	R: 660 nm ± 10	RE: 735 nm ± 5	NIR: 790 nm ±10
<b>TEAX TC2.0 FLIR TAU2 640</b>	TIR: 7.5 - 13.5 $\mu$ m (1 $\mu$ m=1000 nm)			

Note that these spatial and spectral characteristics are absolutely included and considered in the TRL assessment of one camera concerning its ability for assessing a particular physiological parameter. In one classic case, the Normalized Difference Vegetation Index (NDVI), a standard multispectral index, is often used as a point of reference for the general assessment of crop vigor (here vigor considered as a combination of both health and biomass – the more green vegetation present, the higher the NDVI value).

The signal measured by NDVI takes advantage of infrared radiation at approximately 840 nm, just outside of the range of visible light that is detected by RGB cameras; however, even though the difference that it measures between reflected red and infrared radiation is stronger than differences measured in the visible by RGB cameras, the spatial resolution is much smaller. In fact, RGB cameras with their much higher spatial resolutions have often shown to be better, due to its much higher spatial resolution, at assessing general vigor better than basic multispectral cameras, and at a much lower cost.

## C. Unmanned Aerial Vehicle (UAV) Platforms

Currently most modern UAVs fall into two major categories – “fixed wing” and “multirotor.” Fixed wing aircraft are smaller versions of normal planes, composed of a long central body with two large wings on either side equipped with flaps and a vertical tail wing equipped with rudders that provide it with both lift and steering capacities.

Both the flaps and the tail wing rudders are used for steering and minimize yaw (skewed flight), pitch (latitudinal rotation) and roll (longitudinal rotation). Weight distribution may be compensated somewhat using rudders and flaps, but this can make flight much less efficient and as such sensor integration requires more careful consideration.

On the other hand, multirotor UAVs are modifications of modern helicopters. While helicopters often have only one main rotating wing, UAVs more commonly rely on 4-12 separate rotors for providing both lift and steering. Much of this is due to the energy distribution when using batteries for propulsion, but it also includes improvements in stability, redundancy and ease of control. More widely distributed rotors provide increased stability in lighter aircraft while also providing for more lift capacity using smaller, more efficient motors.

The same design allows for more simplified control by using different motor speeds for both rotation and directional movements. In order to compensate for the motor rotation effects, the motors alternate in their spinning direction and as such, increasing the rotation of alternate motors will allow the aircraft to rotate in place while increasing power in two motors side by side will result in lateral movement without any rotation.

In general, sensor integration is not as sensitive in multirotor UAVs compared to fixed wing UAVS since the motors may increase their revolutions in order to compensate directly for minor imbalances in weight distribution; however, this may result in increased motor fatigues and reduced mobility and should be avoided. In general terms fixed wing aircraft are able to fly farther but require more space for take-off and landing and have much higher minimal forward speed requirements.



Multirotor UAVs offer the benefit of vertical take-off and landing with stationary flight capabilities, but with a reduced flight time capacity and area coverage. As such, fixed-wing are more amenable to covering larger areas at lower spatial resolution remote sensing observations (due to higher altitude flights to compensate for image capture and the forward motion requirements), while multirotor UAVs are more flexible and better for covering smaller areas with more detailed spatial resolution remote sensing. The tentative recommendation is to use fixed wing aircraft for precision agriculture and multirotor UAVs for phenotyping (perhaps except in the case of very large phenotyping programs).

## **D. Software**

In the case for software for processing UAV data, there are usually three major parts, 1) image pre-processing including any sensor specific requirements for data calibration or correction, 2) image processing including mosaicking, and 3) image data extraction, which may actually either be directly from the calibrated original image data from (1), from the calibration image mosaic produced in (2), or as part of an advanced hybrid process in which the mosaic image data is used to link back to the original raw data. In the case of image data pre-processing this is almost always supplied by the manufacturer with the purchase of the imaging sensor and will include the steps necessary to use their proprietary techniques for producing calibrated data and exporting it into a standard format. In the case of (2) mosaicking, if this step is chosen, options are limited to just a few Structure-from-Motion implementations – VisualSfM, Pid4D and Agisoft. On the other hand, for (3) data extraction, the options extend to any of a large number of image processing software from free and open-source to commercial packages, with dedicated phenotyping tools available as plugins for FIJI, QGIS, as two examples. This step (3) is by far the most time-intensive.

# Data capture

## A. Basics before going to the field

### 1. Environmental Conditions

- i. **Wind:** In general, the ideal environmental conditions for capturing UAV phenotyping data will be those both amenable to UAV flight (e.g. low wind), but also amenable to quality image capture. Many UAV manufacturer recommendations include UAV flight capacities in 30-50 km/hr winds but flying over 20 km/hr winds will have a severely detrimental effect on image data quality (as well as flight time!).
- ii. **Sunlight:** Image capture should be conducted preferably with clear skies without clouds (or partly cloudy, with at least some large patches of clear skies, but in that case, it may be possible to fly “between clouds” but with great caution towards the cloud shadows). Rapid variations in incoming light intensity, composition, and quality due to either thick cumulus clouds or thin cirrus clouds will render the data from different plots incomparable and therefore useless for the purpose of crop phenotyping.
- iii. **Precipitation:** The UAV should not be flown in the rain. Some UAVs are slightly rain resistant, but most sensors are not waterproof, so you risk total loss of equipment even flying in very light rain. Moreover, image data should not be collected while the crop is wet as it greatly affects the data quality (multispectral and thermal data become completely useless), whether from rain or recent irrigation.

### 2. Time of Day

- i. It is most recommended to collect all sensor data, including RGB, multispectral and thermal images within +/- two hours of solar noon in order to reduce shadow artifacts and other effects related to more oblique incident sun angles.
- ii. RGB digital imaging may be done at any time of day in overcast conditions with only diffuse or indirect light (and thus good contrast but no shadow effects); this can also

produce high quality data because relatively large RGB sensors contain advanced light and white balance adjustment algorithms.

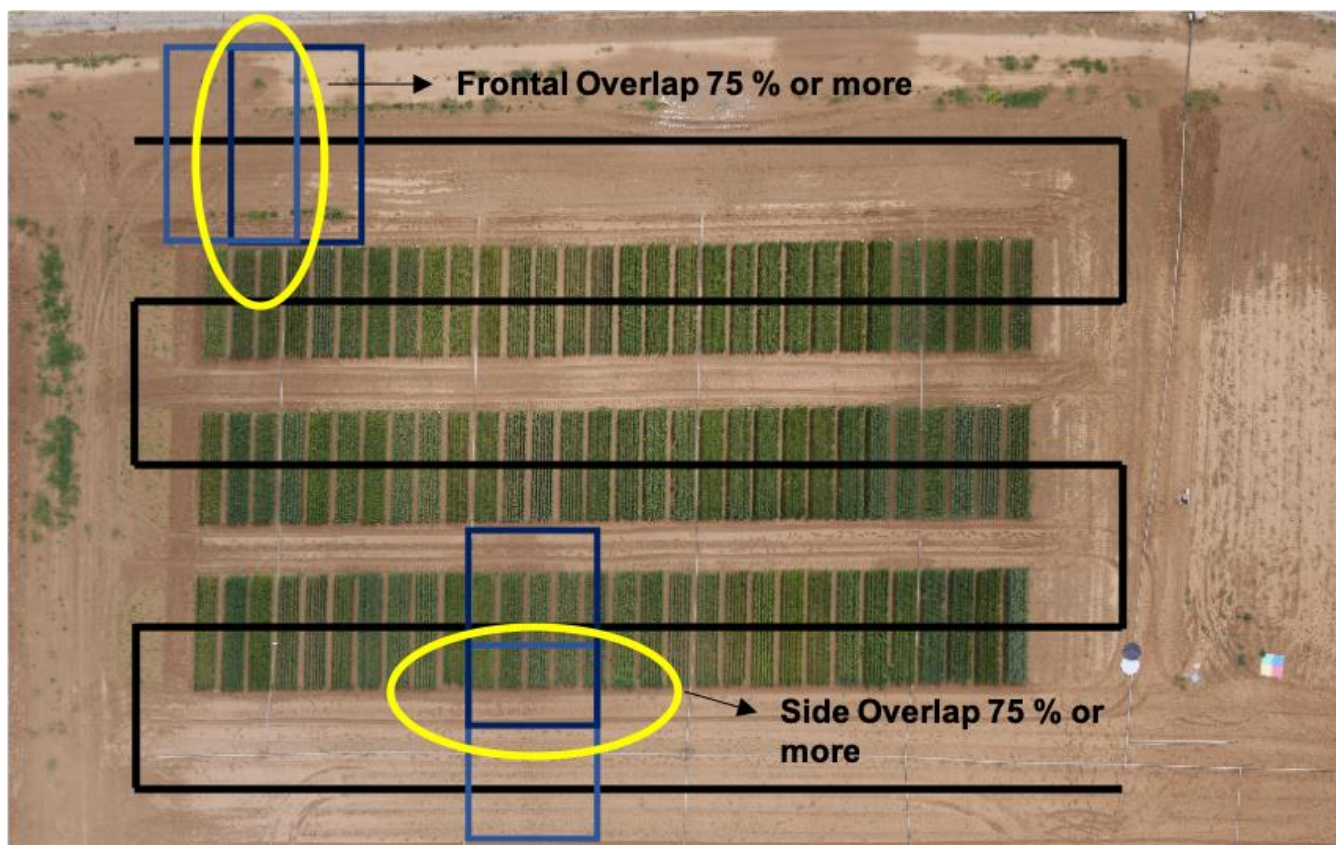
- iii. Some sensors may make sense to fly more than once a day, as they are associated with crop physical parameters that exhibit diurnal fluctuations and thus provide different signals between morning and afternoon, such as canopy temperature, sun-induced fluorescence or even the Photochemical Reflectance Index (PRI), which tracks the xanthophyll cycle compound zeaxanthin.

## **B. Airborne data capture during a UAV flight**

### **1. UAV Flight Planning**

#### **1.1. Planning flight lines for the UAV, whether manual or programmed flight**

- i. UAV phenotyping flight plans should be planned with a minimum of 75% overlap in capture between images along a flight line and 75% overlap between the flight lines (**Figure 1**). This is very important in facilitating the rest of the SOP workflow. When considering the automatic creation of orthomosaics using Structure from Motion (SfM), as detailed in the Data Processing chapter, anything less than 60% will result in model failure and produce gaps, while upwards of 90% may benefit the construction of 3D models, such as in the estimation of plant height as in Part C1. An extra 5-10% overlap should be added for gusty winds.



**Figure 1.** Flight line diagram for a small 72 plot (24x3) 0.25 ha phenotyping study to include extension along of the flight lines beyond the area of study and additional flight lines on each edge in order to include an additional buffer area around the study area. Frontal Overlap (sequential images along a flight line) and Side Overlap (between images of different flight lines) is also indicated.

- ii. UAV flight lines should include an additional 25% beyond the study area in order to ensure full overlapping image coverage for the study area. This should include extra image overlap as mentioned above in case some image pre-filtering may be necessary (as in gusty wind conditions that may cause blurry or oblique images) or additional images captured of calibration panels during take-off and landing if there is not space to place the calibration panels within 10-20 m of the landing platform.

- iii. Image ground spatial distance (GSD) for a specific sensor and UAV flight altitude above ground level (a.g.l.) is a straightforward calculation, which can be completed using several available online aides, such as the Pix4D GSD calculator (<https://support.pix4d.com/hc/en-us/articles/202560249-TOOLS-GSD-calculator>) or others which may include full flight planning support and are often provided with the UAV on purchase (**Table 3** and **Table 4**).

**Table 3.** Example of calculations for Panasonic Lumix GX7 16 MP RGB camera with images at 4592 x 3448 pixels and a Micro 4/3 17.3 mm sensor, showing the lens focal length (FR) and flight requirements for maintaining a 1 cm Ground Spatial Distance (GSD, size of pixels on the ground) at 100, 50 and 25 m for 75% overlap between images (calculated as  $0.75 \times DH = 25.5$  meters). Altitude is above ground level (a.g.l.), DW is the dimension of the width of the resulting image on the ground, DH is the dimension of the height of the image on the ground.

Altitude (m)	FR (mm)	GSD (cm)	DW (m)	DH (m)	Flight Speed (m/s)	Image Frequency (s-1)
100	38	~ 1	46	34	12.5 (or 25)	2 (or 1)
50	19	~ 1	46	34	12.5 (or 25)	2 (or 1)
25	9.5	~ 1	46	34	12.5 (or 25)	2 (or 1)

**Table 4.** Ground Spatial Distance (GSD, pixel dimensions on the ground) calculations flights at 25m, 50m and 100m altitudes for different sensors of typical characteristics for their category. Altitude is above ground level (a.g.l.).

Camera	Sensor size (mm)	Focal Length (mm)	Image size (pixels)	GSD (25m a.g.l.)	GSD (50m a.g.l.)	GSD (100m a.g.l.)
<b>Panasonic Lumix GX7</b>	17.3 x 13.0	20	4592x3448	0.47 cm	0.94 cm	1.88 cm
<b>Airinov Multispec 4C</b>	4 x 6 x 4.8	4	1280x960	2.93 cm	5.86 cm	11.72 cm
<b>TEAX TC 2.0 FLIR TAU2 640</b>	10.88 x 8.7	19	640x512	2.24 cm	4.47 cm	8.95 cm

- iv. Do not change the viewing direction of the multi-rotor UAV during image acquisition, such that all of the images looking in the same direction; this will help to both facilitate the mosaicking process as well as reduce sun-angle artifact difference between images.
- v. Flight lines need to be prepared and adjusted as needed in order to ensure safety with regards to any electricity line, poles, trees, etc. that may obstruct the UAV flight or cause interference with the signal transmission or affect the internal sensors (especially electrical lines). The height of any obstacles should be measured and, if there is any doubt, consider flying at a safer altitude (keeping the image characteristics by adjusting the focal length or accepting either an increase or loss in GSD).

## 1.2. Preflight preparation of the flight area

- i. Prior to UAV deployment, it is necessary to establish a take-off and/or landing platform in a location that is free of hazards and close to the study area. In this sense the vertical take-off and landing capacities of multirotor UAVs, with vertical take-off and landing (VTOL) capacities, provide a major advantage over fixed wing UAVs (fixed-wing UAVs are on the other hand more energy efficient in flight and can cover larger areas).

- ii. UAV landing platforms should be secured flat and stable, and resistant to the wind caused by the UAV, either by using weights or stakes to keep it stationary during the whole of the UAV flight.
- iii. In the case where the end of the planned flight lines results in some distance from the initial take-off platform or the planned flight lines are within 20% of the total estimated maximum flight time, it is advised to establish a secondary/alternative landing location in case battery power ends up being tight in terms of payload and flight times.

### 1.3. Positioning of ground control and calibration targets

- i. Stationary ground control points (GCPs) panels should be marked with an X or large spot to identify their center if used for georeferencing and should be measured with a Geographic Position System (GPS) either before or after the flight, though are not absolutely necessary, especially if the onboard imaging system has minimal integrated GPS measurement capacities (the SfM workflow will still create an image mosaic without GPS data using the visible reference targets). The placement of the GCPs can be fixed for the entirety of the field season or placed and measured before each flight. At least 10 GCPs should be placed per hectare of trial. GCP should be positioned at the perimeters of the study area in positions clear of vegetation or any viewing obstruction.
- ii. Smaller calibration panels made of very precise Labsphere Spectralon, Teflon, or other materials are preferably measured immediately before take-off and then stored before flight in order to ensure cleanliness and consistency between measurements. These are often included with specific multispectral sensors and are required for sensor specific image pre-processing. Commercially available professional photography color panels, such as the XRite Color Checker Passport (<https://xritephoto.com/colorchecker-passport-photo>) may also serve in the same way for RGB camera calibrations.
- iii. Larger in-flight radiometric calibration targets should be positioned prior to flight and remain stationary until all flights have been completed. Radiometric calibration targets should be targets of a known reflectance but can be of any material as long as its spectral reflectance has been measured (synthetic fabric, wooden panels with thick layers of



different colored paints, etc.). They can be round or square but should measure at least 50 cm in diameter or per side. Radiometric targets should be positioned at least 5m from the nearest plot and 10m from the UAV landing platform (to reduce exposure to dust) and surrounded by flat and bare soil area (**Figure 2**).

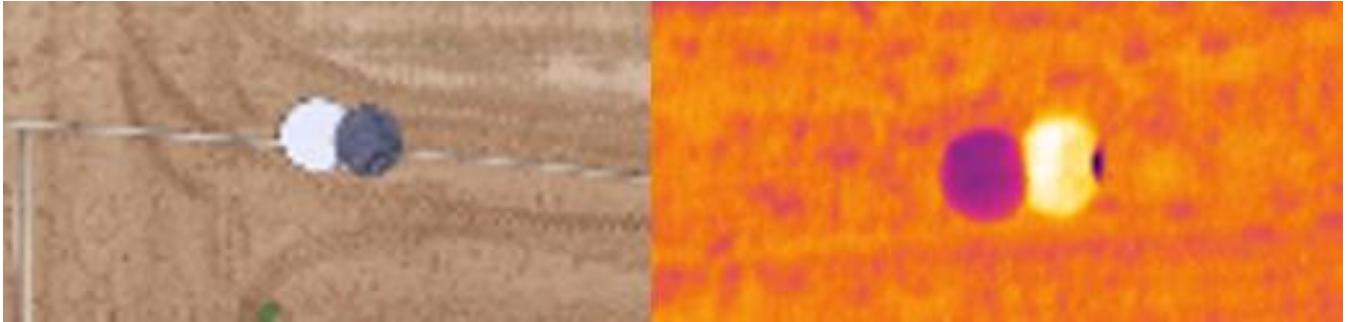


**Figure 2.** Example field preparation for a small 72 plot (24x3) 0.25 ha phenotyping study demonstrating the placement of Geographic Control Points (GCPs), additional reference panels, radiometric calibration panel, and black and white thermal calibration panels with respect to the landing platform and trial.

- iv. White and black targets are preferred for making RGB camera sensor white balance corrections and are also useful for temperature calibration if placed 1-2 hours prior to the UAV flight as they will provide marked temperature differences above and below the crop temperature. In the specific case of temperature calibration panels, it may also be



of interest to capture empirical calibration data at the start and end of the flight or at the end of each flight line if the flight lines are quite long, as this will change by the minute (**Figure 3**).



**Figure 3.** Black and white photographic panels for RGB sensor white balance corrections, shown in the same orientation in a thermal image, with the white panel showing much cooler (darker color) temperatures compared to the black panel's hotter (bright color) temperature.

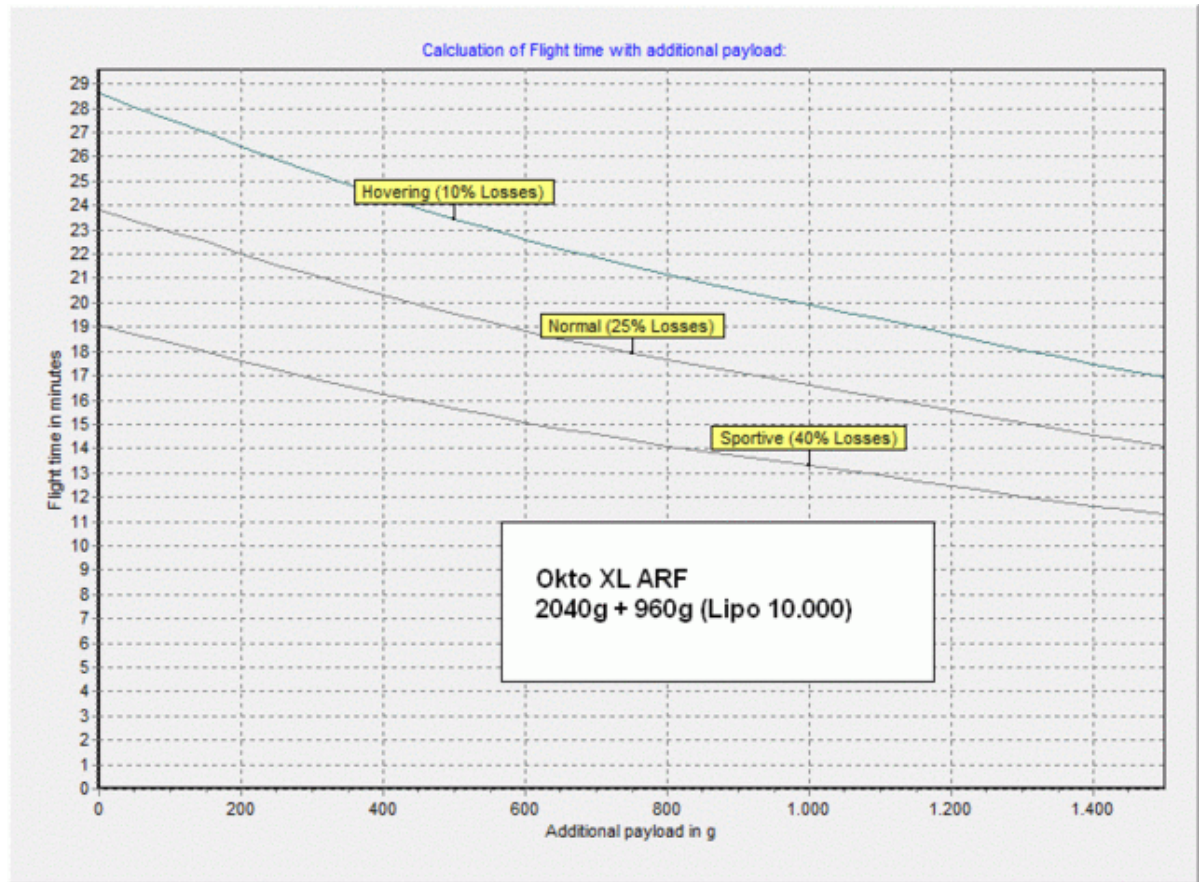
- v. Additional small square “visual reference” panels of bright color (ideally anything but green or brown) should be distributed throughout the study area, especially where GCPs are not present or needed in the overlapping regions between the flight lines. These will aid in improving the automatic steps of the image orthomosaic when using the Structure from Motion (SfM) protocols as they will provide additional tie points between overlapping images and improve the automatic image alignment. There is no need for these to be measured with a GPS or spectrometer as they serve their purpose by simply appearing in the images and aiding the automatic image mosaicking algorithms.

#### 1.4. Testing UAV flight time capacities for different sensor payloads

- i. Establishing maximal flight times for each different payload is very important. In this way one may plan flight line capacities for each payload based on the maximum UAV speed (calculated from the desired image data overlap and the sensor data capture frequency). Many flight planning software includes simulation and/or calculation of total estimated

flight times. Include at least twice the time (4-6 minutes) for landing compared to take-off (2-3 minutes).

- ii. Before each field campaign each year (even for the same UAV and payload as UAV LiPo batteries lose capacity each field season and should be tested yearly), test the flight time for a specific payload: take-off and hover over the landing site until the UAV battery warning sounds, and then land again and record the maximum flight time. Planned flights should include a 20-30% margin of error for safety.
- iii. Remember that even moderate winds near the maximum recommended wind speeds may reduce total UAV flight time by up to 40%, depending on the payload (**Figure 4**). One UAV may fly successfully for years, but will not survive even one major crash, not to mention the potential for damage to the sensor payload!



**Figure 4.** Example payload and wind effects on flight times of the Mikrokopter Okto XL eight rotor UAV with an MTOW of 5kg, weighing 2040g with two LiPo batteries weighing an additional 960g ([http://gallery3.mikrokopter.de/tech/FlightTime\\_OktoXL\\_ARF\\_2\\_EN](http://gallery3.mikrokopter.de/tech/FlightTime_OktoXL_ARF_2_EN))

- iv. The UAV batteries should be fully charged before going to the field as they require a stable energy source for charging and, as they may heat up while charging, need to “rest” for approximately 1 hour between charging and use in the UAV. Check your battery manufacturer specifications for other LiPo safety considerations. One spare set of batteries per planned flight is highly recommended.

## 2. Sensor control and data capture frequency

- i. The preferred method of sensor image data capture control is standalone and separate from the UAV system at the required calculated frequency or higher. Preferentially program the sensor separate from the UAV flight to capture data continuously during the entire flight at the maximum frequency and at the sensor's maximum quality level. Images captured during take-off and landing can easily be filtered afterwards. This ensures data capture with sufficient data capture only then depends on selecting a flight speed moderate enough in order to ensure more than adequate image overlap. Most sensors are designed to capture data at every 1-3 seconds for flights at a speed of 10 m/s at heights of 25m or greater a.g.l. See the above Tables in the section on flight planning for detailed examples.
- ii. Sensor control may also be available through the UAV system or flight planning software, either based on a time frequency similar to described above, or even a spatial frequency based on the UAV GPS. In the case of sensors that are wholly integrated with the UAV the time frequency control is preferred over GPS sensor trigger options.
- iii. When using the integrated UAV flight and sensor control for repeatedly flying the same location with the same payload, these flight plans can be carefully developed and saved either on the PC or the UAV controller memory, depending on the UAV model. Training for using these features is specific to each UAV and flight planner model.
- iv. Even if using an external high precision GPS model with the option to connect to a sensor in order to acquire greater accuracy image by image georeferencing information, it is best to have the imaging sensor send the signal to capture geographic location information to the GPS and not have the image sensor be controlled to trigger image capture by the location information of the GPS system.

### 3. Data storage and documentation

#### 3.1. General best practices for data storage

- i. It is absolutely essential to establish standardized data storage and documentation protocols that match with the existing field data protocols of a specific phenotyping research center or institution. This will aid in the integration of the UAV phenotyping data with the other data sources.
- ii. The raw sensor data should always be saved and backed up in a secure location along with established minimal metadata, such as but not limited to UAV flight plan (UAV model, flight altitude, flight speed and time, sensor payload, pilot, etc.), date, time, and meteorological conditions (temperature, sky/cloud conditions, etc.). Often this is placed in an institutional storage facility.
- iii. Subsequent data processing levels are often later stored in project specific repositories or personal storage areas as the manipulations resulting from calibration, correction, segmentation, and extraction calculations may not be universal in terms of the techniques applied and their targeted use.

#### 3.2. Model solution for UAV data storage (and processing)

The “Pheno-Copter” platform was developed at CSIRO by a team led by Scott Chapman<sup>1</sup> as an integrated image capturing package and a common workflow to guide researchers in acquiring and processing ground level and low-altitude aerial images of vegetation and derive or predict key variables associated with biophysical and count statistics. The project was a collaborative effort between the former CSIRO ICT Centre and CSIRO's former Plant Industry division The Autonomous Systems Program focused on the development of avionics including the sensor payload for an unmanned helicopter - the *Phenocopter*. (redacted from <https://confluence.csiro.au/display/ASL/Phenocopter> and currently maintained active at <https://phenocopter.csiro.au/> or see <https://people.csiro.au/C/S/Scott-Chapman> for more details), including low-altitude remote sensing for applications in agriculture, sensor payload: high-res LWIR, RGB, NIR

vibration isolated cameras, cameras synchronously triggered by flight computer and images recorded on-board at full resolution, mission planning and image analysis software optimized for applications in agriculture, and for more information: [MDPI paper](#).<sup>1</sup>

---

<sup>1</sup>Chapman, S. C., Merz, T., Chan, A., Jackway, P., Hrabar, S., Dreccer, M. F., ... & Jimenez-Berni, J. (2014). Pheno-copter: a low-altitude, autonomous remote-sensing robotic helicopter for high-throughput field-based phenotyping. *Agronomy*, 4(2), 279-30

# Data processing

## A. Post-UAV flight data check

- i. Check UAV image data, preferably in the field before departure in order to ensure data integrity and complete data capture covering the whole of the area of interest. Additional data may still be added at this stage if required. Note that many new UAV sensors are susceptible to errors in acquisition and the use of integrated systems may sometimes result in data loss if depending on GPS coordinates for triggering data capture by the sensors. Visually inspect images for sufficient overlap at 75%.
- ii. Filter unnecessary images prior to initial data storage and documentation, such as any images taken during take-off or landing that are not at the correct altitude of acquisition or do not specifically pertain to the planned image orthomosaic covering the study site. Deleting any accidentally blurry (potentially due to UAV movements or wind gusts) or out-of-focus image is only possible when the image overlap has been acquired closer to 90% in excess of the minimum recommendations of the Part B1 SOP on data capture. Otherwise the loss in image overlap will inhibit orthomosaic creation.

## B. Data quality check by sensor type

### 1. RGB sensors

Preferably filter out any images that have obvious visual defects, such as if image data is blurry when captured during gusty wind conditions or the UAV platform was not entirely stable during flight. In some cases a slightly blurry image may aide in image alignment during the orthomosaic process but then should be later removed off during the creation of the actual orthomosaic image (data quality flagging rather than filtering).

### 2. Multispectral sensors

In the case of many multispectral images that use camera-filter arrays, it is necessary to know the distance between the target and the camera at the time of data capture, in this

case the UAV flight altitude above ground level (a.g.l.), for the next step. The image array will be aligned and combined as a single multi-band image. In this case, if using multispectral flights at different altitudes (as may help in creating properly aligned yet higher resolution multispectral orthomosaics), it may be necessary to pre-process the different altitude overflights first with the sensor-specific software and then combine in only in the orthomosaic process.

### 3. Thermal sensors

Thermal imagers may provide data in any number of different formats as there are not only several different thermal camera manufacturers, but also different providers that are making modifications and add-ons directed as the UAV sector for specific recording, streaming, and data quality needs.

In the specific case of thermal images, because of the sensitivity of the thermal sensor reading to the sensor temperature (which may often change during a flight), the radiometric calibration needs to be applied before building the orthomosaic, because the data will often be provided in radiometrically calibrated centi-Kelvin within a 16-bit image using an undisclosed internal algorithm. Manufacturer specific software should be used to export the image into maximal resolution frames at either 16-bit or 32-bit centi-Kelvin or Celsius formats. Many thermal cameras do not provide full radiometric data or raw digital number (DN) data.

## C. Orthomosaic generation and georeferencing using the structure from motion pipeline

### 1. General pipeline

This can be done with several different software platforms but requires an advanced processing workstation or alternatively by using an online, cloud-based server service.

#### i. Commonly used software:

- Agisoft Photoscan ([http://www.agisoft.com/pdf/photoscan-pro\\_1\\_4\\_en.pdf](http://www.agisoft.com/pdf/photoscan-pro_1_4_en.pdf))



- Pix4D Mapper (<http://www.pix4d.com/>)
- Vision SfM  
([https://d32ogoqmya1dw8.cloudfront.net/files/getsi/teaching\\_materials/high-rez-topo/visual\\_sfm\\_tutorial.pdf](https://d32ogoqmya1dw8.cloudfront.net/files/getsi/teaching_materials/high-rez-topo/visual_sfm_tutorial.pdf))
- DroneMapper (<http://dronemapper.com/>)

iii. Recommended 3D modelling workstation specifications are as follows, per Agisoft Photoscan (<http://www.agisoft.com/downloads/system-requirements/>):

- Advanced Configuration, expect to need upwards of 64 GB of RAM in order to be able to complete mosaics of hundreds of images (+1 ha)
- CPU: Octa-core or hexa-core 3.0Ghz models or higher
- Motherboard: models with 8 DDR4 or DDR3 slots and at least 1 PCI Express x16 slot
- RAM: DDR4-2133 or DDR3-1600, 8 x 8 GB (64 GB total)
- GPU: Nvidia GeForce GTX 980 Ti or AMD RX570 or better

## 2. Camera alignment

The software finds the camera position and orientation for each photo and builds a sparse point cloud model. There are two options:

- Generic:** preselection mode the overlapping pairs of photos are selected by matching photos using the Scale Invariant Feature Transform (SIFT) algorithm, which automatically detects corners and reference points within the image and uses those to align the images. Generic preselection means that the images will be assumed to be sequential, but the alignment will not be aided by the use of georeferencing data.
- Reference:** the overlapping pairs of photos are selected based on the measurement camera locations based on georeferencing data. Note that this does not necessarily mean that alignment will be automatically perfect, as the requirement for highly overlapping images remains the same.

### 3. Import georeferencing data and the Ground Control Point (GCPs)

- i. Some sensors have built-in georeferencing data associated with each image taken during the UAV flight, but this may need to be connected manually to the individual images if not already preset, for example, as part of the EXIF of a JPG image or as part of the image tag of a GeoTIFF. This may be imported into the orthomosaicking software, often in a CSV format, and will be used as an aide to image alignment as well as producing an automatically georeferenced orthomosaic image.
- ii. Using GCPs: these points are evenly distributed around the orthomosaic image. Each program uses different ways to import the points in the mosaic, follow the step in each manual. The quality of the GCPs that are added after the camera alignment process will represent the final georeferencing precision of the orthomosaic and the associated 3D model measurements (plant height, etc.).

### 4. Build geometry or point cloud

In this step, there are several different outputs before the orthomosaic can be exported:

- i. The Sparse Point Cloud is a very low-resolution 3D point cloud that is produced in the process of the Image Alignment as detailed above.
- ii. The Dense Point Cloud is the next step, which uses the image alignment to populate the 3D model with many more points in a 3D cloud using automatically detected tie points and image geometries.
- iii. The 3D Mesh is the triangulation interpretation of the Dense Point cloud that detects the outer surface edge and defines it as the 3D surface of the image.
- iv. The Textured Mesh is then the combination of the image data with the 3D Mesh, which now finally starts to resemble an actual colored 3D model of the image target.

## 5. Digital 3D models as 2D images

In this step, there are several different outputs before the orthomosaic can be exported:

### i. Digital Surface Model (DSM)

Orthomosaicking a set of images with or without georeferenced using the SfM pipeline will produce a 3D surface model, which can be then be used to create a composite orthomosaic image (image with the effects of terrain and UAV flight movements removed) or filtered to produce a separation between terrain (see DEM) and vegetation (see DVM).

The DSM is the only surface model necessary in order to be able to produce the ortho-corrections for exporting an orthomosaic, from which can be extracted different vegetation indexes and specific traits.

Acquiring a digital surface model in the off-season before the crop vegetation emerges will also provide a better digital elevation model (DEM) and there a better baseline for creating a digital vegetation model (DVM), but is not completely necessary if field access for UAV flights is limited, as described below.

### ii. Digital Elevation Model (DEM)

Orthomosaicking a set of images with or without georeferencing can be used for 3D model of the terrain under the vegetation. In order to build the DEM, it is necessary to work on applying filtering to the Dense Point Cloud that is produced as part of the creation of the 3D model in SfM in order to classify the points that are associated with the ground in the model and not vegetation. The resulting points are then used to produce the DEM surface by triangulation, with the associated parameters are generally based on field terrain parameters:

- a. **Maximum Angle** (degrees) – The maximum ground inclination within the specified cell size below. For flattened phenotyping fields, this value should be quite low (i.e. 10-15 degrees)
- b. **Maximum Distance** (linear spatial distance, m) – The maximum vertical distance between any two points on the ground or the maximum vertical variation of any two close ground points.
- c. **Cell Size** (linear spatial distance, m) – The limitation in size of the above parameters, as it determines the size of the moving window for the filtering of dense cloud points based on the above.

Alternatively, if the data can be collected for the phenotyping area of study in the off-season after the field has been prepared for the trial, but before the vegetation emerges, then this is preferred and can be used as the base DEM for the calculation of the digital vegetation model (see DVM below).

### iii. **Digital Vegetation Model** (3D models nDSM or DVM = DSM-DEM)

In some cases, it may be possible to use the actual SfM orthomosaicking software to produce the Digital Vegetation Model, as the difference from the DSM and the DEM, resulting in a vegetation model with the terrain effects removed.

Some specialty software, such as CloudCompare (<https://www.danielgm.net/cc/>), may not include the full SfM pipeline for creating the orthomosaic, but may offer more advanced filtering options for creating the DEM from the DSM as well as the resulting DVM. In that case it is possible to instead export the georeferenced dense point cloud from the SfM software to be subsequently processed.

In some SfM software, it is preferred or necessary to export first the DSM complete model and the DEM terrain only model, perform a smoothing or quality check, and then create the difference image in the image processing software of choice for post processing.

## 6. Export the orthomosaic, DSM, DEM and/or the DVM.

Exporting the orthomosaic image from the SfM (Agisoft, Pix4D, DroneMapper, VisualSFM or other post processing software such as CloudCompare), preferably a lossless format for data analyses, such as TIFF in 16-bits or in 8-bit only if you are certain that this is the maximum quality that your sensor produces. One exception for data may be RGB data, which is by nature normally 24-bit (as in 3 channels of 8-bit data  $3 \times 8 = 24$ ), but they should also be exported as TIFF rather than JPG as it is still lossless and will not retain its image quality characteristics (it decompresses and then recompresses every time when it is opened and loses information).

There is a mathematical way to link between the orthomosaic and the original pictures, in a way that data are extracted from the original images taken from the UAV with no loss in the resolution and without any blending effects. This is preferred if the user is able to correctly achieve the requisite input data quality metrics and manipulate the associated processing code - the GPS precision can be considered roughly equivalent to the exact precision of the resulting 3D model when phenotyping plans include estimating plant height or relying on a Geographic Information System (GIS) for the extraction of micro-plot data, but otherwise, if relying only on the spectral information of the image data and semi-automatic extraction of the micro-plot data, then the high precision GPS requirement may be considered optional:

- i. The original images from the UAV should each be individually georeferenced with a differential GPS system to a precision of preferably 1-3 cm x, y and z.
- ii. The orthomosaic 3D surface model should furthermore be geographically realigned to within the same precision using a grid of at least 30 GCPs per hectare with a precision of +/-1 cm x, y, and z.
- iii. The field trial individual micro-plots need to be reproduced in GIS as either individual shapefiles or one single multipart shapefile with a similar precision of +/- 1 cm x, y and z.
- iv. The exporting algorithm can then use the geospatial information to retrace the coordinates on the orthomosaic through the 3D vectors of each image captured in order

to extract the original image data based on its orthomosaic alignment in the SfM software.

## D. Image Radiometric Calibration

### 1. RGB sensors

- i. Black, Grey and White calibration targets need to be used to correct any white balance issues. This is related to the fact that light conditions vary between flights and affect the reflectance albedo. Therefore, the data to be processed for feature extraction should preferentially first be corrected against a common radiometric reference.
- ii. Special colorimetric panels (normally smaller in size, such as the XRite passport) need to be placed besides the take-off platform, at a minimum distance from both the plots and the take-off platform, surrounded by bare land, to check for color consistency by recording images before and after a flight.

### 2. Multispectral sensors

- i. The specific flight information will need to be provided to the sensor manufacturer's processing software along with images of the specific calibration target for the different sensor wavelengths. The use of a separate non-sensor specific calibration targets may also be used for empirical line calibration, which can be applied after pre-processing or even after the production of the study area image orthomosaic.
- ii. Requirements for specific radiometric calibration parameters for the specific wavelengths of each sensor should be considered separately as well as a sensor-specific alignment between the different imaging sensors.
- iii. The integration of an incident light sensor (ILS) may provide real time radiometric calibration and may or may not also require the use of a specific spectral calibration target. In some cases, the ILS is not band-specific, but a global incidence of light sensor and then a specific calibration panel is needed to calibrate each band to the global ILS in order to provide precise reflectance values.

### 3. Thermal sensors

- i. For basic thermal radiometric calibration, the same empirical line calibration techniques may be deployed using the same Black, Grey and White calibration panels that may be of use for controlling radiometric calibration and albedo or white balance effects of the RGB and the Multispectral sensors.
- ii. For best results with thermal calibration, position the Black, Grey and White panels an hour before the planned thermal flights, such that some time is provided for them to acquire a temperature difference: the Black panel should quickly be hotter than the vegetation, the Grey panel close to or slightly warmer than the vegetation temperature, and the White panel should be close to or slightly cooler than the vegetation temperature.
- iii. As the temperature of vegetation can change quite quickly and, furthermore, the sensitivity of uncooled Vox microbolometer TIR sensors depends on the actual temperature of the sensor, it may make sense to plan for an individual image based advanced TIR calibration by using measurements of TIR calibration panels at the start and end of the data acquisition flight and then use the time decay between the two calibration measurements to apply individual image temperature calibrations before orthomosaicking



## References:

1. Chapman, S.; Merz, T.; Chan, A.; Jackway, P.; Hrabar, S.; Dreccer, M.; Holland, E.; Zheng, B.; Ling, T.; Jimenez-Berni, J. Pheno-Copter: A Low-Altitude, Autonomous Remote-Sensing Robotic Helicopter for High-Throughput Field-Based Phenotyping. *Agronomy* **2014**, *4*, 279–301, doi:10.3390/agronomy4020279.
2. Haghghattalab, A.; González Pérez, L.; Mondal, S.; Singh, D.; Schinstock, D.; Rutkoski, J.; Ortiz-Monasterio, I.; Singh, R. P.; Goodin, D.; Poland, J. Application of unmanned aerial systems for high throughput phenotyping of large wheat breeding nurseries. *Plant Methods* **2016**, *12*, 1–15, doi:10.1186/s13007-016-0134-6.
3. Jin, X.; Liu, S.; Baret, F.; Hemerlé, M.; Comar, A. Estimates of plant density of wheat crops at emergence from very low altitude UAV imagery Remote Sensing of Environment Estimates of plant density of wheat crops at emergence from very low altitude UAV imagery. *Remote Sens. Environ.* **2017**, *198*, 105–114, doi:10.1016/j.rse.2017.06.007.
4. Kefauver, S. C.; Araus-Serret, I.; Vergara-Díaz, O.; Bort, J.; El-Haddad, G.; Nieto-Taladriz, M. T.; Aparicio, N.; Araus, J. L.; Diagonal, A.; Engineer, S.; York, N. Challenges and bottlenecks in UAV phenotyping. In IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium 2018 Jul 22 (pp. 8240-8243). IEEE.
5. Holman, F. H.; Riche, A. B.; Michalski, A.; Castle, M.; Wooster, M. J.; Hawkesford, M. J. High throughput field phenotyping of wheat plant height and growth rate in field plot trials using UAV based remote sensing. *Remote Sens.* **2016**, *8*, doi:10.3390/rs8121031.
6. Duan, T.; Zheng, B.; Guo, W.; Ninomiya, S.; Guo, Y.; Chapman, S. C. Comparison of ground cover estimates from experiment plots in cotton, sorghum and sugarcane based on images and ortho-mosaics captured by UAV. *Funct. Plant Biol.* **2017**, *44*, 169–183, doi:10.1071/FP16123.
7. Deery, David M., Greg J. Rebetzke, Jose A. Jimenez-Berni, Richard A. James, Anthony G. Condon, William D. Bovill, Paul Hutchinson, Jamie Scarrow, Robert Davy, and Robert T. Furbank. "Methodology for high-throughput field phenotyping of canopy temperature using airborne thermography." *Frontiers in plant science* 7. **2016**: 1808. doi.org/10.3389/fpls.2016.01808.

# Data extraction

## A. Trimming the orthomosaic to micro-plots

### 1. Manual

Cutting the ortho-mosaic using any software that has the necessary tools to make polygons, also referred to as Regions of Interest (ROIs) and cut the large orthomosaic into micro-plots (micro-raster images). This technique depends on the number of plots in total and may not be time efficient.

### 2. Semi-automatic

Using an ortho-mosaic with or without geo-reference, there are different software for this work that can help in replicating the trial design and providing a grid of plots that can then be adjusted as needed:

- a. The MosaicTool is a plugin for the FIJI, a free open-source image processing software (<https://fiji.sc/>) that includes tools for creating the micro-plot grid as well as extraction and post-processing (<https://integrativecropecophysiology.com/software-development/mosaictool/>).
- b. The BreederMap that is a similar plugin of the QGIS (<https://github.com/jajberni/BreederMap/wiki>).
- c. The ArcGIS buffer tool that is a plugin of the ArcGIS software, though not open access (<https://pro.arcgis.com/es/pro-app/get-started/install-and-sign-in-to-arcgis-pro.html>).

### 3. Automatic

Using an orthomosaic with precise georeferencing detail and in combination with equally precise GIS vector data for each plot, the data extraction may be done automatically. In this case, some specific code for running this function is needed, often in R, GDAL or Python code as part of a GIS, depending on the software that is used by the agency or user; the processing code will export the image data from the orthomosaic using the micro-plots spatial

information to either create micro-raster plot level images or directly process the data coming from the UAV orthomosaic or even, if available, the original UAV images. Before to import the ortho-mosaic is necessary:

- i. Import the specific GCPs and the list of the point of GPS from the camera and GPS.
- ii. Manually correct the micro-plot position accuracy when super-imposed over the georeferenced orthomosaic image in a GIS, if needed, adjusting for any field trial errors in the micro-plot rectangular shape and dimensions or by re-projecting the orthomosaic image to match the more precise field trial ground layout.
- iii. When the visual assessment of the GIS micro-plot vectors and the orthomosaic is acceptable export both versions from the GIS software.
- iv. Run the R, GDAL or Python code for extracting the micro-plot rasters from the image orthomosaic.

## **B. Extraction of micro-plot information**

### **1. Export the raw data and then calculate vegetation indices**

- i. Save the mean of each band RGB sensors and Multispectral sensors from each micro-plot and go to MS Excel, R or another data matrix style database management software.
- ii. Do the calculation of the different indices manually using simple mathematical equations by hand for all of the desired indices, which may include some examples (**Table 5**).

### **2. Export the data and calculate vegetation indexes using dedicated software tools**

- i. The MosaicTool is a plugin of Fiji (<https://fiji.sc/>) and is used for trimming the orthomosaic, and also has three tabs, one is for RGB sensors the name is “Breedpix” development for the University of Barcelona and CIMMYT where calculated different vegetation indices through three color spaces (HIS, CIEI-Lab and CIE-Lub), one is the “NGRDI TGI Macro” calculate these two indices between RGB channels and the last one

for Multispectral sensors is “MultiSpectral Macro” where use the combination of different bands to calculate vegetation indices some examples are in the last table.

- ii. In QGIS and ArcGIS there is a Raster calculator in batch that calculates the different indices using the RGB bands or multispectral.

**Table 5.** Example vegetation indices and their calculations showing the different multispectral image band wavelengths (R) for reflectance followed by the wavelength number in nm.

Index	Equation	Wavelengths	Reference
Normalized Difference Vegetation Index (NDVI)	$(R790-R660)/(R790+R660)$	Red, NIR	Rouse et al., 1979
Soil Adjusted Vegetation Index (SAVI)	$(R790-R660)/(R790+R660+L)*(1+L)$	Red, NIR	Huete et al., 1988
Optimized Soil-Adjusted Vegetation Index (OSAVI)	$((1+0.16)*(R790-R660))/((R790+R660+0.16))$	Red, NIR	Rondeaux et al., 1996
Renormalized Difference Vegetation Index (RDVI)	$(R790 - R660)/((R790 + R660)^{1/2})$	Red, NIR	Roujean and Breon, 1995
Normalized Difference Red-Edge Index (NDRE)	$(R735-R670)/(R735+R670)$	Red, Red-edge	Horler et al., 1983
Normalized Green Red Difference Index (NGRDI)	$(R550-R660)/(R550+R660)$	Green, Red	Daughtry et al., 2000

### 3. Using specific code

Develop different code examples from R, GDAL or Python where the code processes all the data, starting with the orthomosaic, followed for the plot level data extraction and finally calculating the different vegetation indices of interest and exporting the data all at once.

An example of some specific codes for multispectral and thermal imaging for batch processing the micro-plot processing is given in the following case study by the University of Barcelona.

# Case study: UAV phenotyping on <1000 plots at the University of Barcelona

## Authors:

Shawn C. Kefauver

M. Luisa Buchailot

Jose Luis Araus

(University of Barcelona, Spain)

Remote sensing data capture was completed in this example using RGB, multispectral, and thermal sensors mounted on an Oktokopter XL 6S eight rotor UAV at two separate growth stages. All images were acquired at an altitude above ground level (a.g.l. of 30 m). The RGB and multispectral sensors were mounted and flown for data capture at close to 2:00 pm (solar noon) local time in order to maximize signal quality (sun angle and intensity) and minimize potential crop shading effects in the acquired images. The thermal camera data was captured at 4:30 pm local time in order to capture the moment of maximum temperature differences and possible variability in heat or water stress between the study plots.

Data capture with each sensor was programmed at maximum quality and frequency for the duration of the UAV flights. All cameras were programmed for continuous capture for the duration of the flight while the UAV was flown manually at the specified elevation of 30 m over the study site in 6 flight lines in order to ensure sufficient image overlap for mosaicking in post-processing. The Panasonic Lumix GX7 RGB sensor captures data at a rate of one image every 2 seconds during each flight, resulting in 162 images and 148 images recorded at 30 m altitude during the first and second flights, respectively. The Multispec 4C multispectral sensor acquires 10-bit images also at a rate of one image every 2 seconds during flight, providing 186 and 192 images for each flight. The multispectral images were calibrated using images acquired pre-flight of a sensor-specific calibration panel and during flight using an incident light sensor to detect any changes in light quality during the flight. The FLIR TAU2 640 thermal sensor, other the other hand, due to its lower resolution, is able to acquire images at 20 frames

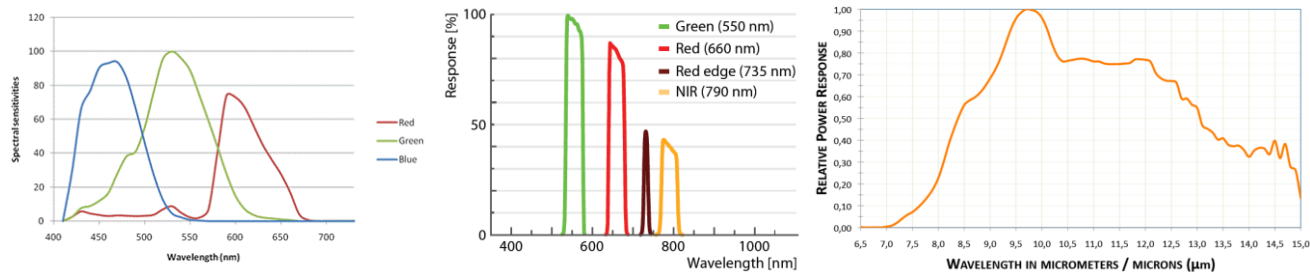
per second in 640/512 pixel full resolution video, resulting in approximately 1,671 and 1,964 images, respectively.

## Sensors used for image data capture

For this project the University of Barcelona has selected the most adequate sensors for the a pilot project on the use of UAV remote sensing for crop selection of the target crop of maize abiotic stress phenotyping, including a high resolution semi-professional quality digital camera (referred to as the RGB camera as it captures visible light), a four band multispectral sensor that include two bands outside of visible light in the red-edge and near infrared regions, and a thermal imager. The spectral sensitivity details for the multispectral and thermal cameras are further detailed as follows.

Camera	Sensor size (mm)	Sensitivity	Lens (mm)	Resolution (pixels)
<b>Lumix GX7</b>	17.3 x 13.0	8 bit RGB Live MOS	20	4592x3448
<b>Multispec 4C</b>	4 x 6x4.8	10 bit G, R, RedEdge, NIR CMOS	4	1280x960
<b>FLIR TAU2 640</b>	10.88x8.7	Uncooled VOx Microbolometer	19	640x512

**Table 6.** Camera and sensor details for the RGB, multispectral and thermal cameras.



**Figure 5.** Graphical representations of the spectral sensitivities of a typical RGB natural color camera using a standard CMOS sensor with a Bayer RGB filter (a, left), the AIRINOV Multispec 4C multispectral sensor band sensitivities using a standard CCD sensor and specific waveband filters (b, center, [www.airinov.fr](http://www.airinov.fr)) and the FLIR Tau2 640 Uncooled VOx Microbolometer unfiltered image sensor sensitivity used for thermal data capture (, right, [www.flir.com](http://www.flir.com)).

**Table 7.** Spectral sensitivity details of the AIRINOV Multispec 4C and FLIR Tau2 640 sensors.

Camera	Band 1	Band 2	Band 3	Band 4
Multispec 4C	G: 550 nm	R: 660 nm	RE: 735 nm	NIR: 790 nm
FLIR TAU2 640	TIR: 7.5 - 13.5 $\mu$ m			

\*G=Green, R=Red, RE=Red Edge, NIR=Near Infrared, TIR=Thermal Infrared



## UAV image data pre-processing

All of the available images from each sensor and flight were used in the production of the orthomosaics images with the commercial software Agisoft Photoscan Professional using standard processing methods. The resulting image mosaic ground spatial distance pixel resolutions were estimated at 3.35 mm per pixel for the RGB images, 2.89 cm pixel resolution for the multispectral images, and 1.13 cm pixel resolution for the thermal images, being approximately the same for each flight date.

The image mosaics were then processed using the University of Barcelona - Integrative Crop Ecophysiology Group proprietary “MosaicTool” software designed by Dr. Shawn C. Kefauver to segment the mosaics into plot sized images of exactly the same ground spatial area for each plot and each image sensor corresponding to the size of the study design plots. The segmentation process is semi-automatic in that the initial grid of plot images is produced automatically, then these plots were individually adjusted to compensate for melon plant shifts due to berm and wind effects. This decision was based on discussions between the Syngenta and University of Barcelona teams while in the field.

While at first appearing to be a daunting task, the intelligent design of the “MosaicTool” software allowed for the manual micro-adjustment of 882 plots for three flights in two visits in approximately one hour of technician time per flight. The resulting plot level images included just over 120,000 pixels of data per plot for the RGB images, more than 4,500 pixels per plot from the multispectral camera, and over 5,000 pixels of data per plot from the thermal images.

The image analyses were then conducted at the plot level using the suite of processing code built in the “MosaicTool” and using a set of custom macro codes written specifically for application in this project for melons. The RGB images were analyzed using our modified version of the Breedpix code originally developed and published by Jaume Casadesus of IRTA in Lleida, Spain (Casadesus et al. 2007). Our modifications include the Crop Senescence Index, a combination index of the Green Area (GA) and Greener Green Area (GGA) green pixel

quantifications that has higher values with increased crop senescence. This includes average plot values in three different color spaces (HIS - Hue, Saturation and Intensity, CIE-Lab and CIE-Luv) and the image segmentation green pixel-based analyses GA, GGA and CSI derived from the Hue component of the HIS color space.

The four-band AIRINOV multispectral sensor allows for the calculation of a number of vegetation indexes related to crop vigor, all of which include components of chlorophyll content (health) and crop biomass (growth), together understood as vigor in this context. The average values of the four separate reflectance bands are presented as C1-C4 for the G, R, RE, and NIR bands, respectively. The subsequently calculated vegetation indexes include the Normalized Difference Vegetation Index (NDVI), Soil-Adjusted Vegetation Index (SAVI), Optimized Soil-Adjusted Vegetation Index (OSAVI), Renormalized Difference Vegetation Index (RDVI), Normalized Red-Edge Index (NDRE), and the Normalized Green Red Difference Index (NGRDI), calculated as follows in

**Table 8.**

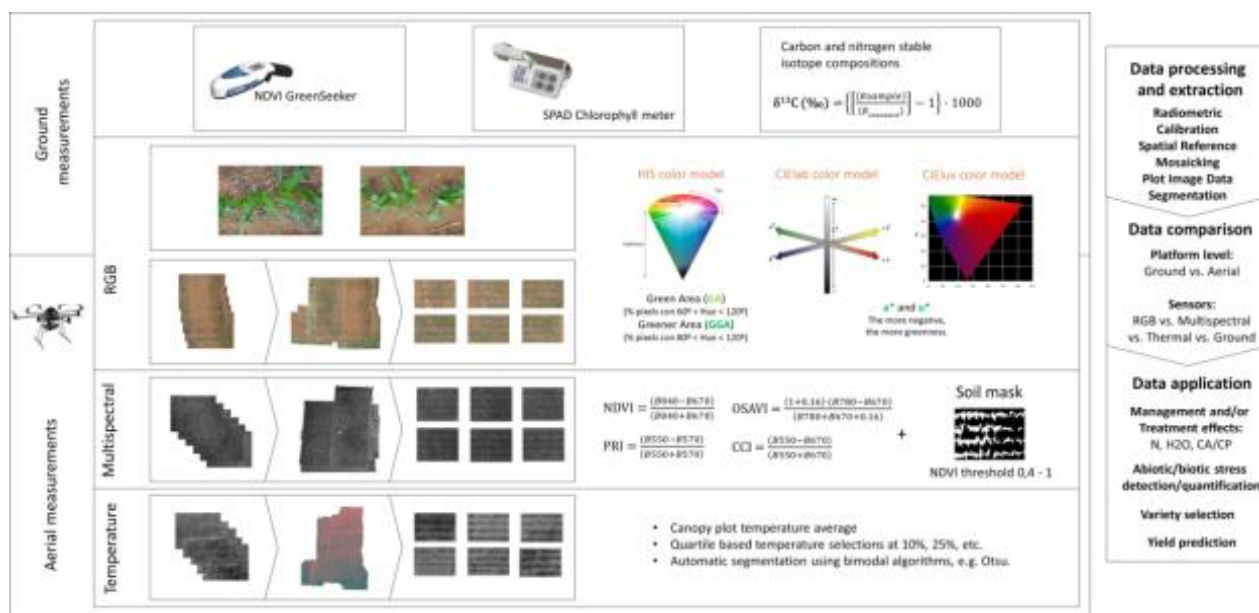
**Table 8.** Details on the calculation, wavelength regions and original scientific reference for the multispectral indexes as calculated from the AIRINOV Multispec 4C NDVI/NDRE sensor.

Index	Equation	Wavelengths	Reference
Normalized Difference Vegetation Index ( <b>NDVI</b> )	$(R790-R660)/(R790+R660)$	Red, NIR	Rouse et al., 1979
Soil Adjusted Vegetation Index ( <b>SAVI</b> )	$(R790-R660)/(R790+R660+L)*(1+L)$	Red, NIR	Huete et al., 1988
Optimized Soil-Adjusted Vegetation Index ( <b>OSAVI</b> )	$((1+0.16)*(R790-R660))/((R790+R660+0.16))$	Red, NIR	Rondeaux et al., 1996
Renormalized Difference Vegetation Index ( <b>RDVI</b> )	$(R790 - R660)/((R790 + R660)^{1/2})$	Red, NIR	Roujean and Breon, 1995
Normalized Difference Red-Edge Index ( <b>NDRE</b> )	$(R735-R670)/(R735+R670)$	Red, Red-edge	Horler et al., 1983

Normalized Difference Index ( <b>NGRDI</b> )	Green	Red	$(R550-R660)/(R550+R660)$	Green, Red	Daughtry et al., 2000
---	-------	-----	---------------------------	------------	--------------------------

## MosaicTool image microplot data processing details

Due to the high resolution of the acquired multispectral and thermal images, we were able to develop custom code that includes the calculation of whole plot values, but also accurately separates the vegetation from the soil background and provides both the vegetation indexes and the surface temperature of only the vegetation dominated pixels within each study plot. In the case of the thermal data, we have also provided both the mean and the median values of the whole plot and only the vegetation. These are notes with the `_plot` and `_veg` distinction for the multispectral indexes and the same with the addition of `_mean` and `_median` for the thermal details, as requested, are additionally provided in the final data matrix. The separation between vegetation and non-vegetation in the multispectral images was done using the standard reference NDVI value of 0.4 to separate pixels that were clearly vegetation from the rest of the image. This value was also used to calculate the “Vegetation Area” parameter at the start of the multispectral section of the data matrix. In the case of the thermal images, the matter was a bit more complex. First the values were manually filtered to exclude unrealistic values outside of the range of 5 to 75 degrees C (resulting from the image processing mosaicking and segmenting chain) and then this range of realistic values was segmented for vegetation using the bimodal-specific thresholding method by Otsu (1979).



**Figure 6.** Overview of integration of ground level measurements and UAV data processing from RGB, multispectral and thermal data analyses, with standardized processing from UAV to individual plot-level image data extraction. Optional inclusion of image segmentation/classification techniques may be processed on UAV and ground level image data. (Figure modified from the graphical abstract of Garcia-Romero et al., Remote Sensing, 2018 <http://www.mdpi.com/2072-4292/10/2/349>)

## References

1. Casadesús J, Kaya Y, Bort J, Nachit MM, Araus JL, Amor S, et al. Using vegetation indices derived from conventional digital cameras as selection criteria for wheat breeding in water-limited environments. *Ann. Appl. Biol.* 2007;150:227–36.
2. Rouse JW, Hass RH, Schell JA, Deering DW. Monitoring vegetation systems in the great plains with ERTS. *Third Earth Resour. Technol. Satell. Symp.* 1973;1:309–17.
3. Huete A. A soil-adjusted vegetation index (SAVI). *Remote Sens. Environ.* 1988;25:295–309.
4. Roujean, Jean-Louis, and Francois-Marie Breon. "Estimating PAR absorbed by vegetation from bidirectional reflectance measurements." *Remote sensing of Environment* 51.3 (1995): 375-384.
5. Horler, D. N. H., Mo DOCKRAY, and J. Barber. "The red edge of plant leaf reflectance." *International Journal of Remote Sensing* 4.2 (1983): 273-288.
6. Daughtry C. Estimating Corn Leaf Chlorophyll Concentration from Leaf and Canopy Reflectance. *Remote Sens. Environ.* [Internet]. 2000;74:229–39.
7. Rondeaux G, Steven M, Baret F. Optimization of soil-adjusted vegetation indices. *Remote Sens. Environ.* 1996;55:95–107.
8. Otsu, Nobuyuki. "A threshold selection method from gray-level histograms." *IEEE transactions on systems, man, and cybernetics* 9.1 (1979): 62-66.

## Specific code used

### RGB FIJI Plug-in

The CIMMYT MaizeScanner plugin for FIJI was used to process all of the RGB data from this study, once segmented into individual microplot images. It is freely available at: <https://github.com/sckefauver/CIMMYT>

### Multispectral FIJI Macro

```
// Macro to batch process plot level images cut from the orthomosaic of a multispec4C by AIRINOV
print("\Clear");
print("Macro Multispec4C AIRINOV Multispectral Data Vegetation Indices Calculator Plus");
a = newArray("filename", "VegetationArea", "C1", "C2", "C3", "C4", "NDVIplot", "NDVIveg", "SAVIplot", "SAVIveg",
    "OSAVIveg", "OSAVIplot", "RDVIplot", "RDVIveg", "NDREplot", "NDREveg", "NGRDIplot", "NGRDIveg");
Array.print(a);
//toggle for plugin defined input and output directories
//var input = "$P{batch_input}";
//toggle for testing manually defined input and output directories
var input = "D:\\Temp\\input\\";
var output = "D:\\Temp\\output\\";
var list = getFileList(input);
setBatchMode(true);
run("Clear Results");
function isImage(filename) {
    if(endsWith(toLowerCase(filename), ".tif") ||
        endsWith(toLowerCase(filename), ".tiff") ||
        endsWith(toLowerCase(filename), ".jpg") ||
        endsWith(toLowerCase(filename), ".jpeg")) {
        return true;
    }
    else {
        return false;
    }
}
```

```

};

for (i = 0; i < list.length; i++) {
    if(isImage(list[i])) {
        action(list[i]);
    }
};

selectWindow("Results");
//saveAs("Results", "$P{save_results_file}");
saveAs("Results", output + "ResultsMultispec4C-G-R-RE-NIR.csv");
run("Close");
call("java.lang.System.gc");
setBatchMode(false);
function action(filename) {
    open(input + filename);
    original = getImageID();
    imageId = original;
    title = getTitle();
    imageName = File.nameWithoutExtension;
    bit = bitDepth();
//Split channels for processing and rename files
    selectImage(imageId);
    channels = nSlices;
    run("Stack to Images");
    for(j = 0; j < channels; j++) {
        imageId -= 1;
        selectImage(imageId);
        run("32-bit");
//adjust bit depth of each band depending on original bit rate of image
        if(bit == 8) {
            run("Divide...", "value=256 stack");
        }
        if(bit == 16) {
            run("Divide...", "value=65536 stack");
        }
    }
}

```

```

        rename("C"+(j+1));
    };
//use image math calculator plus functions to calculate the chanel's mean of the whole plot
//C1
    selectWindow("C1");
    run("Set Measurements...", "mean area_fraction redirect=None decimal=2");
    run("Measure");
    c1 = getResult("Mean");
//C2
    if(channels >= 2) {
        selectWindow("C2");
        run("Set Measurements...", "mean area_fraction redirect=None decimal=2");
        run("Measure");
        c2 = getResult("Mean");
    }
    else {
        c2 = "NAN";
    }
//C3
    if(channels >= 3) {
        selectWindow("C3");
        run("Set Measurements...", "mean area_fraction redirect=None decimal=2");
        run("Measure");
        c3 = getResult("Mean");
    }
    else {
        c3 = "NAN";
    }
//C4
    if(channels >= 4) {
        selectWindow("C4");
        run("Set Measurements...", "mean area_fraction redirect=None decimal=2");
        run("Measure");
        c4 = getResult("Mean");
    }

```



```

else {
    c4 = "NAN";
}

// Continue with vegetation area and vegetation indices for plot and veg
// Use image math calculator plus functions to calculate the NDVI of the whole plot and create a soil
mask
    imageCalculator("Subtract create 32-bit", "C4", "C2");
    rename("ResultC4-C2");
    imageCalculator("Add create 32-bit", "C4", "C2");
    rename("ResultC4+C2");
    imageCalculator("Divide create 32-bit", "ResultC4-C2", "ResultC4+C2");
    rename("IMG_NDVI");
// Create soil bed area mask and measure the NDVI of vegetation area
    selectWindow("IMG_NDVI");
    run("Duplicate...", "title=IMG_NDVI2");
    selectWindow("IMG_NDVI2");
    rename("IMG_NDVI_soil");
    selectWindow("IMG_NDVI_soil");
    setThreshold(0,50);
    setOption("BlackBackground", false);
    setAutoThreshold("Default");
    setAutoThreshold("Default dark");
    run("Convert to Mask");
    run("Despeckle");
    run("Invert");
    rename("Soil_mask");
    run("Set Measurements...", "mean area_fraction redirect=None decimal=4");
    run("Select All");
    run("Measure");
    Vegetation_area = getResult("%Area");
    setResult("Type", nResults-1, "Vegetation_area");
    setResult("File Name", nResults-1, filename);
    updateResults();
//save the NDVI value for the average of the whole plot
    selectWindow("IMG_NDVI");

```

```

run("Set Measurements...", "mean area_fraction redirect=None decimal=4");
run("Measure");
NDVI_plot = getResult("Mean");
setResult("Type", nResults-1, "NDVI_plot");
setResult("File Name", nResults-1, filename);
updateResults();
// Use image math calculator plus functions to calculate the SAVI of the whole plot
imageCalculator("Subtract create 32-bit", "C4", "C2");
rename("ResultC4-C2");
imageCalculator("Add create 32-bit", "C4", "C2");
rename("ResultC4+C2");
selectWindow("ResultC4+C2");
run("Add...", "value=0.5");
rename("ResultC4+C2+05");
imageCalculator("Divide create 32-bit", "ResultC4-C2", "ResultC4+C2+05");
rename("ResultC4-C2/C4+C2+05");
selectWindow("ResultC4-C2/C4+C2+05");
run("Multiply...", "value=1.5");
rename("IMG_SAVI");
run("Set Measurements...", "mean area_fraction redirect=None decimal=4");
run("Measure");
SAVI_plot = getResult("Mean");
setResult("Type", nResults-1, "SAVI_plot");
setResult("File Name", nResults-1, filename);
updateResults();
// Use image math calculator plus functions to calculate the OSAVI of the whole plot
imageCalculator("Subtract create 32-bit", "C4", "C2");
rename("ResultC4-C2");
selectWindow("ResultC4-C2");
run("Multiply...", "value=1.16");
rename("Result116C4-C2");
imageCalculator("Add create 32-bit", "C4", "C2");
rename("ResultC4+C2");
selectWindow("ResultC4+C2");
run("Add...", "value=0.16");

```

```

rename("ResultC4+C2+016");
imageCalculator("Divide create 32-bit", "Result116C4-C2","ResultC4+C2+016");
rename("IMG_OSAVI");
run("Set Measurements...", "mean area_fraction redirect=None decimal=4");
run("Measure");
OSAVI_plot = getResult("Mean");
setResult("Type", nResults-1, "OSAVI_plot");
setResult("File Name", nResults-1, filename);
updateResults();
// Use image math calculator plus functions to calculate the RDVI of the whole plot
imageCalculator("Subtract create 32-bit", "C4","C2");
rename("ResultC4-C2");
imageCalculator("Add create 32-bit", "C4","C2");
rename("ResultC4+C2");
selectWindow("ResultC4+C2");
run("Square Root");
rename("ResultC4+C2^1/2");
imageCalculator("Divide create 32-bit", "ResultC4-C2","ResultC4+C2^1/2");
rename("IMG_RDVI");
run("Set Measurements...", "mean area_fraction redirect=None decimal=4");
run("Measure");
RDVI_plot = getResult("Mean");
setResult("Type", nResults-1, "RDVI_plot");
setResult("File Name", nResults-1, filename);
updateResults();
// Use image math calculator plus functions to calculate the NDRE of the whole plot
imageCalculator("Subtract create 32-bit", "C3","C2");
rename("ResultC3-C2");
imageCalculator("Add create 32-bit", "C3","C2");
rename("ResultC3+C2");
selectWindow("ResultC3+C2");
imageCalculator("Divide create 32-bit", "ResultC3-C2","ResultC3+C2");
rename("IMG_NDRE");
run("Set Measurements...", "mean area_fraction redirect=None decimal=4");
run("Measure");

```

```

NDRE_plot = getResult("Mean");
setResult("Type", nResults-1, "NDRE_plot");
setResult("File Name", nResults-1, filename);
updateResults();
// Use image math calculator plus functions to calculate the NGRDI of the whole plot
imageCalculator("Subtract create 32-bit", "C1","C2");
rename("ResultC1-C2");
imageCalculator("Add create 32-bit", "C1","C2");
rename("ResultC1+C2");
selectWindow("ResultC1+C2");
imageCalculator("Divide create 32-bit", "ResultC1-C2","ResultC1+C2");
rename("IMG_NGRDI");
run("Set Measurements...", "mean area_fraction redirect=None decimal=4");
run("Measure");
NGRDI_plot = getResult("Mean");
setResult("Type", nResults-1, "NGRDI_plot");
setResult("File Name", nResults-1, filename);
updateResults();
// Create soil bed area mask and measure the NDVI of vegetation
selectWindow("Soil_mask");
run("Duplicate...", "title=Soil_mask-save");
selectWindow("Soil_mask-save");
run("Set Measurements...", "mean area_fraction redirect=IMG_NDVI decimal=4");
run("Create Selection");
run("Make Inverse");
run("Measure");
NDVI_veg = getResult("Mean");
setResult("Type", nResults-1, "NDVI_veg");
setResult("File Name", nResults-1, filename);
updateResults();
// Create soil bed area mask and measure the SAVI of vegetation
selectWindow("IMG_NDVI");
run("Duplicate...", "title=IMG_NDVI2");
selectWindow("IMG_NDVI2");
rename("IMG_NDVI_soil");

```

```

selectWindow("IMG_NDVI_soil");
setThreshold(0,50);
setOption("BlackBackground", false);
setAutoThreshold("Default");
setAutoThreshold("Default dark");
run("Convert to Mask");
run("Despeckle");
run("Invert");
rename("Soil_mask");
run("Set Measurements...", "mean area_fraction redirect=IMG_SAVI decimal=4");
run("Create Selection");
run("Make Inverse");
run("Measure");
SAVI_veg = getResult("Mean");
setResult("Type", nResults-1, "SAVI_veg");
setResult("File Name", nResults-1, filename);
updateResults();
// Create soil bed area mask and measure the SAVI of vegetation
selectWindow("IMG_NDVI");
run("Duplicate...", "title=IMG_NDVI2");
selectWindow("IMG_NDVI2");
rename("IMG_NDVI_soil");
selectWindow("IMG_NDVI_soil");
setThreshold(0,50);
setOption("BlackBackground", false);
setAutoThreshold("Default");
setAutoThreshold("Default dark");
run("Convert to Mask");
run("Despeckle");
run("Invert");
rename("Soil_mask");
run("Set Measurements...", "mean area_fraction redirect=IMG_OSAVI decimal=4");
run("Create Selection");
run("Make Inverse");
run("Measure");

```

```

OSAVI_veg = getResult("Mean");
setResult("Type", nResults-1, "OSAVI_veg");
setResult("File Name", nResults-1, filename);
updateResults();
// Create soil bed area mask and measure the RDVI of vegetation
selectWindow("IMG_NDVI");
run("Duplicate...", "title=IMG_NDVI2");
selectWindow("IMG_NDVI2");
rename("IMG_NDVI_soil");
selectWindow("IMG_NDVI_soil");
setThreshold(0,50);
setOption("BlackBackground", false);
setAutoThreshold("Default");
setAutoThreshold("Default dark");
run("Convert to Mask");
run("Despeckle");
run("Invert");
rename("Soil_mask");
run("Set Measurements...", "mean area_fraction redirect=IMG_RDVI decimal=4");
run("Create Selection");
run("Make Inverse");
run("Measure");
RDVI_veg = getResult("Mean");
setResult("Type", nResults-1, "RDVI_veg");
setResult("File Name", nResults-1, filename);
updateResults();
// Create soil bed area mask and measure the NDRE of vegetation
selectWindow("IMG_NDVI");
run("Duplicate...", "title=IMG_NDVI2");
selectWindow("IMG_NDVI2");
rename("IMG_NDVI_soil");
selectWindow("IMG_NDVI_soil");
setThreshold(0,50);
setOption("BlackBackground", false);
setAutoThreshold("Default");

```

```

setAutoThreshold("Default dark");
run("Convert to Mask");
run("Despeckle");
run("Invert");
rename("Soil_mask");
run("Set Measurements...", "mean area_fraction redirect=IMG_NDRE decimal=4");
run("Create Selection");
run("Make Inverse");
run("Measure");
NDRE_veg = getResult("Mean");
setResult("Type", nResults-1, "NDRE_veg");
setResult("File Name", nResults-1, filename);
updateResults();
// Create soil bed area mask and measure the RDVI of vegetation
selectWindow("IMG_NDVI");
run("Duplicate...", "title=IMG_NDVI2");
selectWindow("IMG_NDVI2");
rename("IMG_NDVI_soil");
selectWindow("IMG_NDVI_soil");
setThreshold(0,50);
setOption("BlackBackground", false);
setAutoThreshold("Default");
setAutoThreshold("Default dark");
run("Convert to Mask");
run("Despeckle");
run("Invert");
rename("Soil_mask");
run("Set Measurements...", "mean area_fraction redirect=IMG_NGRDI decimal=4");
run("Create Selection");
run("Make Inverse");
run("Measure");
NGRDI_veg = getResult("Mean");
setResult("Type", nResults-1, "NGRDI_veg");
setResult("File Name", nResults-1, filename);
updateResults();

```

```

function calculateMeanChannelVeg(window, col, filename) {
  selectWindow("IMG_NDVI");
  run("Duplicate...", "title=IMG_NDVI2");
  selectWindow("IMG_NDVI2");
  rename("IMG_NDVI_soil");
  selectWindow("IMG_NDVI_soil");
  setThreshold(0,50);
  setOption("BlackBackground", false);
  setAutoThreshold("Default");
  setAutoThreshold("Default dark");
  run("Convert to Mask");
  run("Despeckle");
  run("Invert");
  rename("Soil_mask");
  run("Set Measurements...", "mean area_fraction redirect="+window+" decimal=4");
  run("Create Selection");
  run("Make Inverse");
  run("Measure");
  setResult("Type", nResults-1, col);
  setResult("File Name", nResults-1, filename);
  updateResults();
};

r = newArray(filename, Vegetation_area, c1, c2, c3, c4, NDVI_veg, NDVI_plot, SAVI_veg, SAVI_plot,
  OSAVI_veg, OSAVI_plot, RDVI_veg, RDVI_plot, NDRE_veg, NDRE_plot, NGRDI_plot, NGRDI_veg);
Array.print(r);
selectWindow("Log");
saveAs("Text", output + "MultispectralMacroMultispec4C_Results_LogArray.csv");
//End of the batch for each image
  run("Close All");
};

function calculateMeanChannel(window, col, filename) {
  selectWindow(window);
  run("Set Measurements...", "mean area_fraction redirect=None decimal=4");
  run("Measure");

```



```

    setResult("Type", nResults-1, col);
    setResult("File Name", nResults-1, filename);
    updateResults();
};

```

## Thermal Infrared FIJI Macro

```

//Macro to batch process convert Thermal Infrared TIFFS
//Prepare organized log output
    print("\Clear");
    print("Macro ThermalImageMacroFLIRTau640TEAX-TIR-Kelvin-Celsius-BimodalMeans");
a = newArray("filename", "VegetationPercentArea", "Tplot_mean", "Tplot_median", "Tveg_mean",
"Tveg_median");
Array.print(a);
//batch processing intro setup
//where the TIFF images to be processed are stored
    input = "D:\\Temp\\input\\";
//outputs directory for JPEGs
    output = "D:\\Temp\\output\\";
setBatchMode(true);
run("Clear Results");
//start batching
list = getFileList(input);
for (i = 0; i < list.length; i++)
    action(input, output, list[i]);
function action(input, output, filename) {
//open and get id and base name
    open(input + filename);
//get info from original image
    original = getImageID;
    title = getTitle;
    base = File.nameWithoutExtension;
//make sure you have the right image selected before starting
selectWindow(title);

```

```

//This section will divide the image using an automatic thresholding technique, which should give us
two groups - //vegetation and ground
rename("image");
selectWindow("image");
run("Duplicate...", "title=veg");
run("Duplicate...", "title=TIR");
selectWindow("image");
run("Set Measurements...", "area mean median area_fraction redirect=TIR decimal=4");
setThreshold(5.0000, 75.0000);
run("Create Selection");
run("Measure");
Tplot_mean = getResult("Mean");
Tplot_median = getResult("Median");
selectWindow("image");
run("Convert to Mask");
rename("mask");
run("Create Selection");
//Vegetation Selection
selectWindow("veg");
run("Restore Selection");
setAutoThreshold("Otsu dark");
run("Convert to Mask");
run("Restore Selection");
run("Set Measurements...", "area mean median area_fraction redirect=None decimal=4");
run("Measure");
Vegetation_area1 = getResult("Area");
//Redirect to the original image in order to apply the mask and measure the ground temperature
selectWindow("veg");
run("Set Measurements...", "area mean median area_fraction redirect=TIR decimal=4");
run("Create Selection");
run("Make Inverse");
run("Measure");
Tveg_mean = getResult("Mean");
Tveg_median = getResult("Median");
Vegetation_area2 = getResult("Area");

```

```
selectWindow("TIR");
run("Select All");
run("Measure");
Image_area = getResult("Area");
Vegetation_area = Vegetation_area2/Vegetation_area1;
//save the different measurements in an organized fashion
r = newArray(filename, Vegetation_area, Tplot_mean, Tplot_median, Tveg_mean, Tveg_median);
Array.print(r);
selectWindow("Log");
saveAs("Text", output + "ThermalImageMacro_FLIR_Tau640_Bimodal_Results_LogArray.csv");
//End of the batch for each image
run("Close All");
```