ALMA MATER STUDIORUM - UNIVERSITY OF BOLOGNA

DEPARTMENT OF AGRICULTURAL AND FOOD SCIENCES

CAMPUS OF BOLOGNA

MASTER'S DEGREE IN

PRECISE AND SUSTAINABLE AGRICULTURE

"*DEVELOPMENT OF A PYTHON BASED SYSTEM FOR AUTOMATED APPLE ORCHARD BLOSSOM MAPPING THROUGH THE ANALYSIS OF DRONE ACQUIRED IMAGES*"

Thesis on

Precision Orchard Management

Relatore:

Chiar.mo Prof. Luigi Manfrini

Correlatore:

Dott. Gianmarco Bortolotti

Candidato:

Mirko Piani

Matricola N° 981542

Academic Year 2021 /2022.

“If you can dream it,

you can do it”

Walt Disney.

Index

ABSTRACT

Flower load is one of the earlier indicators of potential yield in fruit orchards. Usually, flower clusters are present on a tree in more than needed quantities for an optimal production. The most used techniques to manage the flower load are manual, mechanical, and chemical thinning. The main issue is to calibrate these techniques on the base of the desired yield. Drone imagery, being able to collect highly detailed information, could offer a solution to automate flower counting since manual flower counting would be too laborious. The main goals of this study were to develop an easy-to-use python program for sudden field interventions focused on both mapping site – specific and small area's flower load. The program requirements were short computing time, high reliability, and full automatization of the process between data collection and information supply. Those were achieved by applying and comparing three methodologies that allow to map apple's flower clusters load at blooming while adapting the quantification method to different training systems (2 – dimensional and 3 – dimensional). A comparison between the flower cluster estimation and the actual cluster load on hundred trees highlighted that mapping flower clusters by binary classification coupled with geoprocessing is more suitable than other similar methods, since it allowed to have a reduction in the processing time of 89 – 99%, an easier and less noise - affected image analysis with R^2 values ranging from 0,76 to 0,85. Moreover, the study highlighted the correlation between site – specific variability and mean local variation ($r = 0.77$, $R^2 = 0.94$) suggesting that to implement a VRT thinning at first the local mean variation can be mapped (requiring 44 s /ha) and secondly a site – specific flower load quantification can be performed according to the training system (3D: 25 min/ha; 2D: 27 min/ha). The proposed methodologies seemed to be able to manage with a high spatial variability.

Keywords— Precise management, thinning, flower load, fruit quality, yield, remote sensing.

1. INTRODUCTION

1.1. IMPORTANCE OF MONITORING THE CLUSTER LOAD

Over recent years, with the development of Precision Orchard Management (POM) techniques the monitoring of blossom cluster load per tree have been more and more important, since it is broadly recognised that crop load management is the single management strategy most influencing farm's annual profits (Robinson et al., 2021). This implies that to improve the production, the farmer should be able to manage the orchard spatial variability before fruit set, in this way it would be possible to forecast the potential crop load and take proper decisions on how to perform as soon as possible a variable rate thinning (Liakos et al., 2017). Early interventions and knowledge about the spatial variability of blossom load are the basis to enhance fruit size, since at the beginning of the fruit growth, the among fruitlets competition for carbohydrates influences fruit's cell division. Consequently, the earlier it is known in which parts of the orchard a higher than needed fruit load is expected to be realised, the earlier we could decide to give priority on thinning those areas to remove the on – going excessive competition which results in smaller fruits (Lakso et al., 1989). Furthermore, flower/crop load varies considerably over the orchard both in terms of initiation and intensity due to i) crop genetics ii) individual-tree crop load in the previous year (Link, 2000), iii) management differences (i.e. thinning, pruning, irrigation, plant protection strategies as protecting covers and bioregulators applications) (Yang and Xu, 2021) and iv) because of stressing environmental conditions (Naor et al., 2006) affecting either fresh tissues integrity (i.e. late frosts) or resources availability. Thus, in field, at a given time it could be find either a condition in which there is high spatial variation in blooming intensity ranging from trees with no flower clusters to trees with excessive load or a condition in which the flowering is homogeneous. According to this, different decisions must be taken on the thinning intensity, switching from a Variable Rate Treatment (VRT) thinning to a homogenous thinning (Manfrini, 2009). Another aspect to consider apart from flower

load is blooming's scalarity, which may bring the plants to reach the target crop load at different times. For example, in 2020, in Voghiera (Ferrara, IT), a late frosts period came at the end of March while the apple cultivar 'Fuji' had already initialised blooming. That resulted in strong flowers necrosis with different impacts on the yield according to flower load, since the effect on high flowered trees was no more than a natural thinning which still allowed the farmer to get the target crop load out of them. Contrarily, on low flowered trees (i.e., those which initialised blooming later and those in the "year – off ") the apparent loss seemed to be the overall potential yield. Later in the season, at the end of April with the restoring of normal environmental conditions, a strong secondary blooming period occurred leading the farmer to get the target crop load even in those trees which apparently lost their whole production. Hence, letting just the trees suffering from biennial bearing disorder with no fruits. Obviously, since apple production is a match between fruit size and quality parameters as colour and sugar content, the revenue coming from late – blooms – originated fruits is lower than the one originated from early – blooms. This is as a consequence of lower fruit quality (Dalhaus et al., 2020), but the gain in revenues still enriches the farmer allowing to reduce the impact of costs on his profit. Therefore, with cropping conditions strongly influenced by climate change, high price and production cost volatility, being able to scrape production from every tree as possible by monitoring blooming and performing a VRT thinning based on the cumulative flower quantity per tree and not according to a time snapshoot of the flower load, seems to be essential for the future of fruit growers' activity.

1.2. TECHNOLOGY TO MAP FLOWER DENSITY

Even if the importance of flower load monitoring is widely recognised and the research trend on it is growing year after year to bring the farm sustainability (Figure 1), the availability for methods

which have potential field application meeting the automation of mapping flower load with ease of use, high reliability, short processing time and cheapness is still limited.

Prior to year 2022, different methods for the estimation of clusters density in fruit orchards were developed starting from georeferenced manual sampling and spatial data interpolation (Manfrini et al., 2009; Teodorescu et al., 2016), following more recently with Machine Learning (ML) algorithms applied both to images shot by rover – embedded cameras (Wang et al., 2020) (Scalisi et al., 2021) and Unmanned Aerial Vehicle's ones (Vanbrabant et al., 2020).

Another proposed technique to estimate the cluster load at field level is based on the multi – level binary segmentation of an RGB orthomosaic (Piani et al., 2021) characterised by trees' top – side view and high spatial resolution (1.02 cm/pixel). In the last cited paper, the Authors concluded stating that the binary segmentation of RGB drone imagery seems to be a valid approach for fostering field application. To put it more simply, it does not require the technology training phase as ML does, it allows with short processing time (8 mins), moderate correlation between manually sampled and estimated cluster load (R^2 = 0.71) and potential low cost technology (drone with RGB camera and

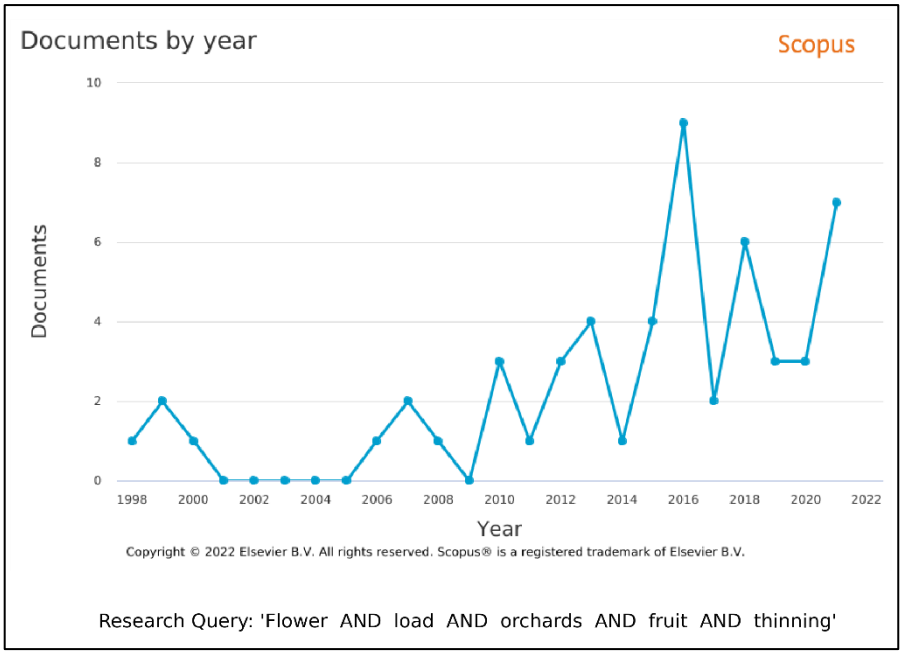


Figure 1. Scopus Literature research

open source GIS software) if compared with effective commercial prototype platforms as "cartographer" (Scalisi et al., 2021). On the other side of the coin, it also has drawbacks as:

- the need for expensive licenced photogrammetry software to stitch the photograms in order to get the orthomosaic that should be analysed.
- the long processing time required (6 hours) by a commonly owned Personal Computer (PC, i3-7020U CPU, 2.30GHz, 4.00 GB RAM) to produce the aerial mosaic.
- considering orchards' heterogeneous lighting conditions, the mosaic multi – photogram composition requires dynamic colour thresholding for image segmentation and proper small objects detection.
- the noise given by soil on image segmentation, then flower detection.
- the need for complete automation of the data analysis and manipulation phases.

When developing a potential new commercial technology, the automation of processes should aim to promote farmers adoption and avoid them to spend time in training themselves on the use of the required software for data analysis and manipulation, since they are most likely to not be familiar with them considering that the European agricultural sector is suffering by ageing of farmers (European Union, 2021) and needs generational renewal. The automation should then provide the main drivers of technology adoption as Ease of Use and Usefulness/Perceived Usefulness (Pierpaoli et al., 2013) in managing a relevant agronomic procedure like flower counting. Thus, supplying a smooth transition to POM techniques, making farmers more inclined to adopt technology besides their own experience along their decision-making process.

1.3. DRONES IN AGRICULTURE

Unmanned aerial Vehicles (UAVs), also known as drones, are aircrafts without any on-board human passenger and are controlled remotely by the pilot who can either modify manually the camera

and flight setting or defining a flight path which the drone could travel by itself. In agriculture drones are becoming more and more popular, since they can be used to inspect either large or difficultly accessible areas in an easier and quicker way in comparison to field scouting (Tripicchio et al., 2015). The UAV has embedded sensors and actuators which communicate with the Central Processing Unit (CPU) allowed with computing power. This characteristic is the reason why drones are uploaded with many algorithms which allow their use in agriculture for many tasks: landscape and crop monitoring, remote sampling, 3D reconstruction, precise delivery of pesticide and Biological Control Agents (BCA), object detection and monitoring (Basri et al., 2021; van der Merwe et al., 2020), fruit picking.

According to the utility purpose, the sensors installed in a drone may vary, but in most cases the main navigation and payload sensors are:

- positioning measurement system (i.e., Global Navigation Satellite System, GNSS), which allows the drone to know its position and move around the landscape following the flight plan.
- Inertial Measurement Units (IMUs), which tells the drone and cameras orientation.
- Cameras (RGB, RGB-D, Multispectral, Hyperspectral, Thermal), which are the “eyes” of the drone, allowing it to sense and to either detect or avoid objects.
- Light Detection and Ranging (LIDAR) is exploited for environment and landscape 3D reconstruction.
- Altimeter, (i.e., barometer, laser altimeter, ultrasonic sensor) which allows the drone to get its flight elevation for image post – processing analysis.

1.3.1. IMAGERY SPECTRAL PROPERTIES

Correspondingly to the type of camera embedded in the drone, the spectral imaging varies into RGB, multispectral, hyperspectral, and thermal. More generally, in terms of spectral resolution, optical cameras are classified from the most to the least sensible as follows:

a) hyperspectral, they sense in 100 – 200 contiguous spectral bands of 5 – 10 nm bandwidth, then they are the most sensible and expensive, and from their application, it is possible to classify objects very precisely based on image thresholding methods.
b) Multispectral, they sense in 5 – 10 bands with 70 – 400nm bandwidth. Usually, the most common spectral wavebands are Red (R), Green (G), Blue (B), Red-Edge (R_{edge}) and Near Infrared (NIR). With this type of camera it is either possible to classify the landcover according to the use of Vegetation Indices (VIs) or to assess if a crop is in optimal or suboptimal health status (Daponte et al., 2019; Gautam and Sarkar, 2020).
c) RGB, they sense just three bands in the visible spectrum (R, G, B), then their spectral resolution is quite low, but considering those bands are the ones in which human eyes can see, RGB cameras become useful for Computer Vision based on human – eye colour perception. Furthermore, RGB cameras, which are the cheapest ones, may be added with Infrared emitters and cameras (RGB-D) to match distance and spectral information which enhanced robots and aircrafts autonomous navigation (Zhang et al., 2019) as well the achievement of true colour Digital Terrain Model (DTM) and Digital Surface Model (DSM).
d) Thermal cameras, which sense the energy emitted by bodies in the thermal region of the spectrum, are used to map surface temperature, crop water status and either humans or animals tracking.

Therefore, digital images, also referred to as raster, are 2D arrays which dimensions depend on the size of the camera sensor. Then as a matrix, an image is composed by a finite number of cells, called “pixels”, which host the spectral information collected by the optical sensor. It should be specified that digital images are composed by multiple channels (or “layers” if referred to in a Geographic Information Systems context) according to the number of bands the camera may sense. As an example, a RGB camera of 21Mpx may produce digital images with three layers of 5120 columns by 4096 rows (image size: 5120 x 4096 x 3), thus 21 million pixels each (Figure 2). The

number of pixels defines both the resolution of the image and the spatial resolution when converting the image into a spatial raster. That is not all, since the resolution of a raster from proximal or remote sensing is a match between optical sensor's size and flight elevation, the effective raster spatial resolution or "Ground Sampling Distance" (GSD) is dynamically determined only by the elevation or Above Ground Level (AGL), since as it can be expected the sensor's size does not change. To calculate the GSD of a survey or in other words, to know to how many ground metres a pixel corresponds, it is possible to apply the following formula:

$$GSD\ (m)\ = \frac{H\ \times\ d}{p}$$

Where H is the distance between the ground and the camera, d is a pixel's side dimension of the optical sensor and p is the focal length (Corradeghini, 2018).

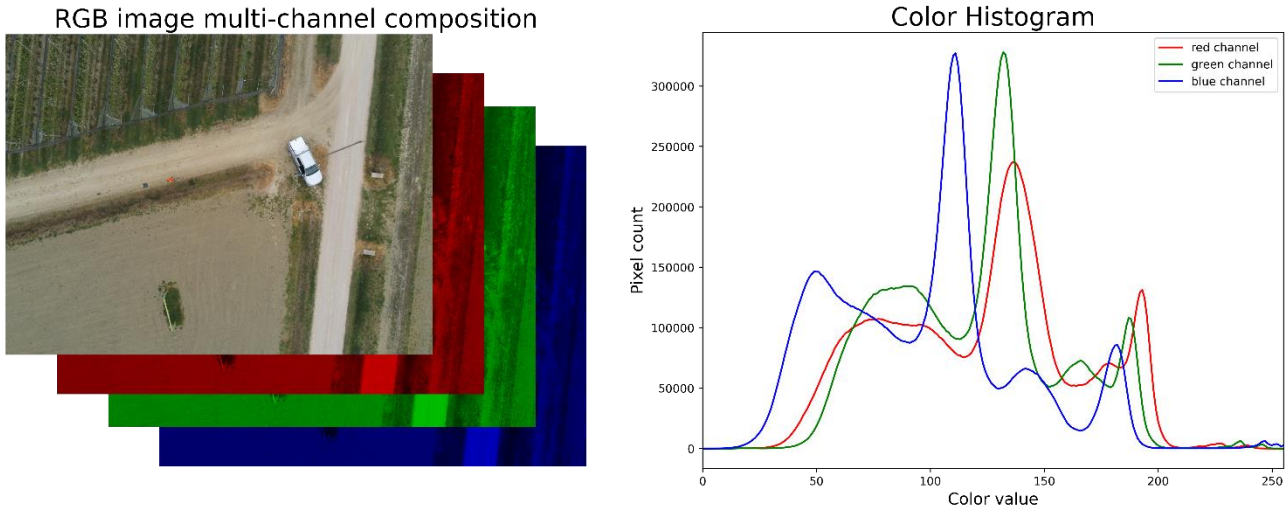


Figure 2. RGB image multi-channels composition and spectral data distribution

1.3.2. IMAGERY METADATA

According to the camera type, the drone may produce images in JPEG (RGB) or TIFF format (multispectral, hyperspectral, thermal). Whatever the file format, images are integrated with the data (metadata) and information collected by the navigation and payload sensors both as Exchangeable

Image File (Exif) and Extensible Metadata Platform (XMP). The latter format is the standard for TIFF and GeoTiff files and it is based on Extensible Markup Language (XML). Generally, Exif stores GPS location, time, and camera settings at the time the image was shot, while XMP stores drone and camera's physical location and direction. Metadata (Figure 3) are important since they store the information needed by photogrammetry programs to process and stich the images together into the orthomosaic.

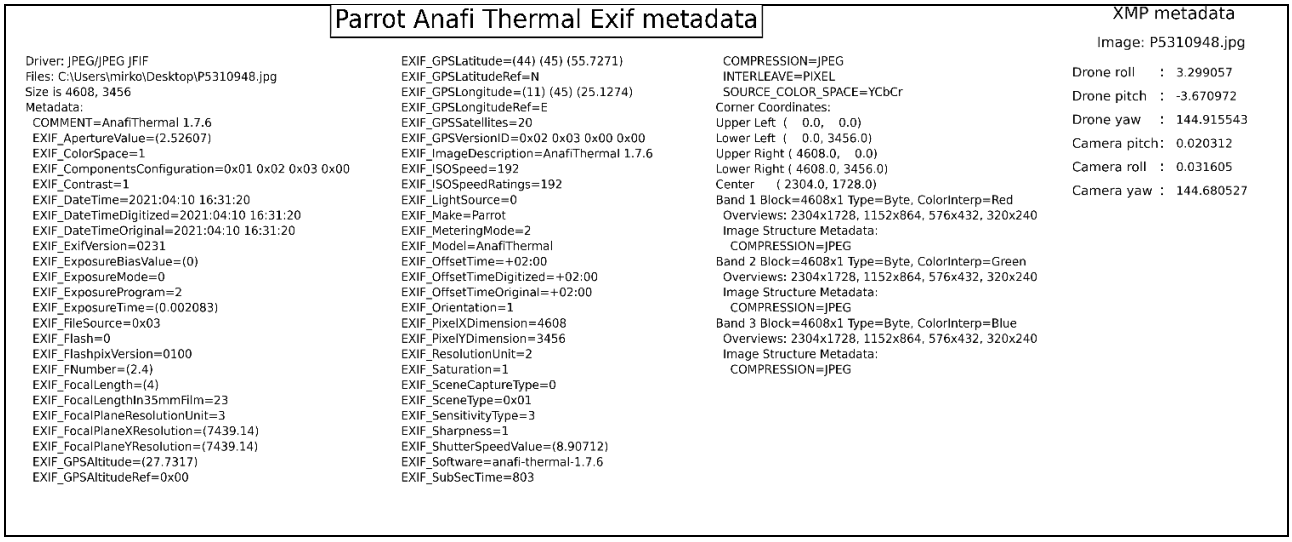


Figure 3. JPEG format metadata

1.4. GEOREFERENCING DRONE IMAGERY

Before an aerial image can be used to support VRT, it is essential to georeferenced its matrix, which means associating to each pixel geographic coordinates (*longitude*, *latitude*, *altitude*) and projection coordinates (*x*, *y*) with the proper coordinate reference system (*crs*).

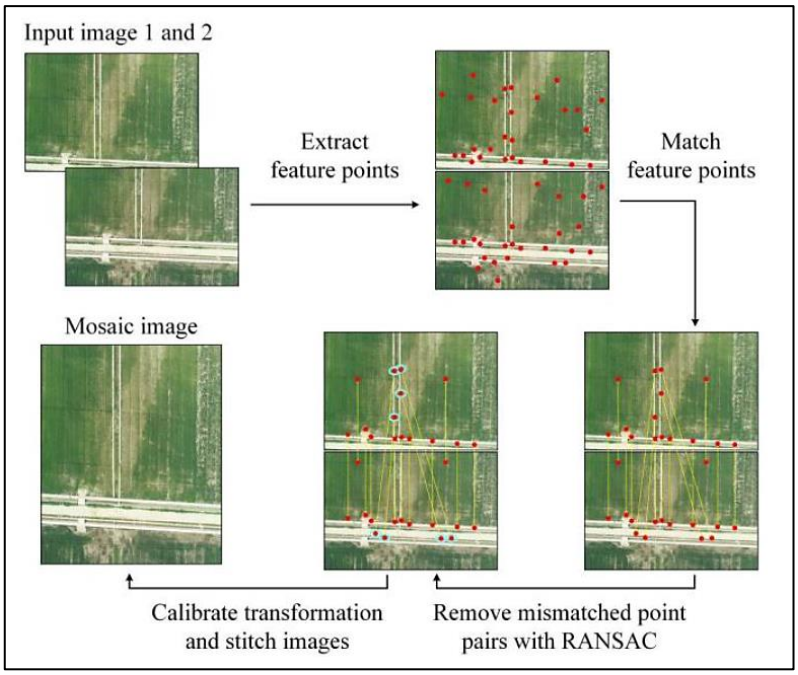


Figure 4. Flowchart of standard scale-invariant feature transform (SSIFT)-based image mosaicking (Zhao et al., 2019)

This process can be performed in several ways, starting from the most adopted strategy based on manual georeferencing relying on Ground Control Points (GCPs), or more recently, following the disruption driven by aerial survey, with either strategies based on direct georeferencing without GCPs (but using navigation and camera metadata) or georeferencing following photogram mosaicking based on feature comparison and stitching as showed in Figure 4 (Zhao et al., 2019). The major problem with direct georeferencing is the error in the estimation of flight elevation (Štroner et al., 2021), since this factor influences the magnitude of the spatial resolution. In Precision Agriculture (PA) high temporal and spatial resolution images are very useful – simply think about flower or pest detection which requires to distinguish objects smaller than 2 cm^2 from the background (Ren et al., 2018) – but mosaicking images taken at low altitudes on large areas is too time consuming (Moran, 1997) as well the collection of GCPs (Xiang and Tian, 2011). Then images geometric corrections are needed to perform the direct georeferencing and reduce the image distortions (Figure 5). A promising technique for keeping high accuracy is based on multiple georeferenced UAV's images mosaiced

together, which outcome is a quickening in the georeferencing and data processing phase. Clearly it represents an advantage in terms of user acceptability (Xiang and Tian, 2011) to support site-specific management.

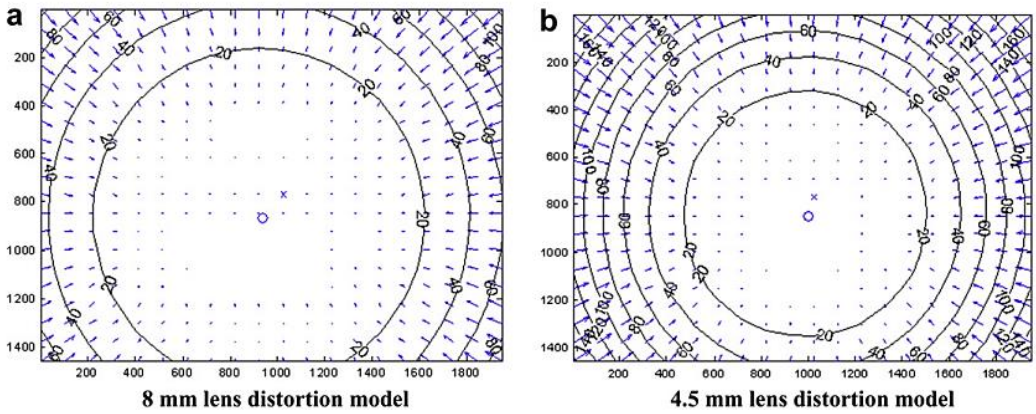


Figure 5. Lens distortion models. (Xiang and Tian, 2011)

1.5. GEOPROCESSING

The term "geoprocessing" refers to a framework and set of tools for processing geographic and related data for spatial and temporal analysis as well variability assessment. Those aforementioned tools could be very useful to automate GIS tasks and quicken geodata analysis. Following, typical tasks of GIS tools are listed:

- operations on datasets,
- change of *crs*,
- rasterization,
- vectorisation,
- georeferencing,
- overlay (intersection, union, exclusion, clipping)
- buffer analysis,
- spatial interpolation with deterministic or geostatistics methods

GIS software like QGIS and GRASS are based on programming languages which like Python and Java relate on libraries supplied by OSGeo (‘Open-Source Geospatial Foundation’) as GDAL/OGR for raster and vector layers manipulation, and Orfeo ToolBox (OTB) which introduces machine learning application to remote sensed data. Those modules can be used as open-source tools to be implemented into python programs to perform inputs data preparation, analysis, documentation, and presentation, thus automating the data analysis and transformation flow which separates inputs from desired outputs. GDAL/OGR library supports many file drivers as JPEG, PNG, TIFF, GTiff (GeoTiff) for raster data as well ESRI Shapefile and GeoJSON file formats for vector data.

In the GIS workflow, spatial data can be either vector layers or raster layers, and according to their nature they fit for different purposes:

- vector layers: they are discrete data; thus, they are represented by points, lines and polygons storing data and geographical/cartographic coordinates according to their location. For example, trees’ locations in orchards may be represented by using points vector layers, while tree rows by lines vector layers and orchard boundaries by polygons.
- Raster layers: they are continuous data, meaning that they consist of a grid of pixels containing specific information as surface reflectance, yield, canopy health, air temperature and so on. It is even possible to get continuous data by interpolation of discrete data in order to get trends on the feature in analysis.

Python as a worldwide spread programming language has the availability for some others modules as GeoPandas (https://geopandas.org/en/stable/) which allow programmers to work on and map vector geodata in an easier way in comparison to OSGeo, and does it by combining other modules as Pandas (https://pandas.pydata.org/) and Shapely (https://shapely.readthedocs.io/en/stable/). The core data structure of GeoPandas is the GeoDataFrame (Figure 6) which consists in a database of indexed features whit their respective geometry (point, line, polygon). Here following there is an example of basic processing with

GeoPandas in Figure 7.

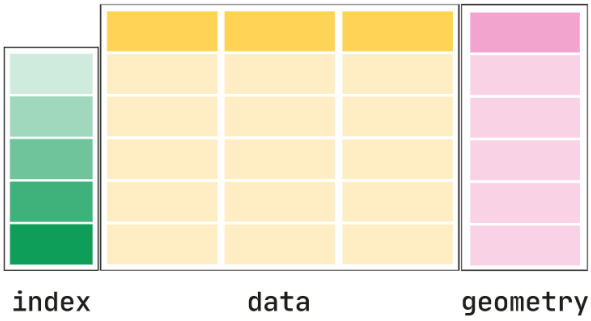


Figure 6. GeoPandas' GeoDataFrame structure

```
In [47]: import geopandas

         # importing vector data
         path_to_data = geopandas.datasets.get_path("nybb")
         gdf = geopandas.read_file(path_to_data)

         gdf
```

Out[47]:

	BoroCode	BoroName	Shape_Leng	Shape_Area	geometry
0	5	Staten Island	330470.010332	1.623820e+09	MULTIPOLYGON (((970217.022 145643.332, 970227....
1	4	Queens	896344.047763	3.045213e+09	MULTIPOLYGON (((1029606.077 156073.814, 102957...
2	3	Brooklyn	741080.523166	1.937479e+09	MULTIPOLYGON (((1021176.479 151374.797, 102100...
3	1	Manhattan	359299.096471	6.364715e+08	MULTIPOLYGON (((981219.056 188655.316, 980940....
4	2	Bronx	464392.991824	1.186925e+09	MULTIPOLYGON (((1012821.806 229228.265, 101278...

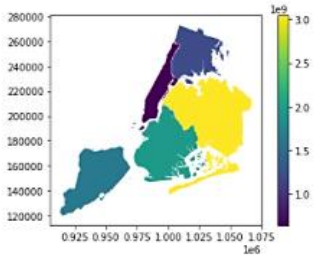
```
In [48]: # adding features to the GeoDataFrame
         gdf["area"] = gdf.area
         gdf
```

Out[48]:

	BoroCode	BoroName	Shape_Leng	Shape_Area	geometry	area
0	5	Staten Island	330470.010332	1.623820e+09	MULTIPOLYGON (((970217.022 145643.332, 970227....	1.623822e+09
1	4	Queens	896344.047763	3.045213e+09	MULTIPOLYGON (((1029606.077 156073.814, 102957...	3.045214e+09
2	3	Brooklyn	741080.523166	1.937479e+09	MULTIPOLYGON (((1021176.479 151374.797, 102100...	1.937478e+09
3	1	Manhattan	359299.096471	6.364715e+08	MULTIPOLYGON (((981219.056 188655.316, 980940....	6.364712e+08
4	2	Bronx	464392.991824	1.186925e+09	MULTIPOLYGON (((1012821.806 229228.265, 101278...	1.186926e+09

```
In [49]: # plotting the GeoData
         gdf.plot("area", legend=True)
```

Out[49]: <AxesSubplot:>




```
In [50]: # making the map interactive
         gdf.explore("area", legend=False)
```

Out[50]:

Figure 7. GeoPandas basic workflow

1.6. SEMIVARIOGRAM

As already mentioned, field conditions are heterogeneous, and the core of PA is to figure out the magnitude of spatial and temporal variability in order to manage them through agronomical decisions characterised either by uniform or variable applications (Florin et al., 2010). To reach this goal, it is essential to collect georeferenced data or to use technology as yield monitors, proximal sensors and remote sensing drones or satellites to automatically sample the field (van der Merwe et al., 2020). Once the data are collected and discrete values are associated with geographical coordinates (*longitude, latitude*) through the georeferencing process, it is possible to analyse the spatial correlation of variables through the semivariogram (or variogram) geostatistical tool. The semivariogram computes the probable degree of dissimilarity among two points separated by a given distance (h - *lag*) by analysing the semivariance (Eq. 1) between all points separated by the defined *lag*.

$$\gamma(h) = \frac{1}{2} \sum_{i=1}^{n(h)} \frac{\left(z(x_i + h) - z(x_i)\right)^2}{n(h)} \qquad \text{(Equation 1)}$$

To get the semivariogram model (Matheron, 1963), the first step is to convert all the geographical coordinates from degrees (GPS in WGS84 Datum) to their respective projections in metres (*x, y*). In this way, it is possible to calculate Euclidean distances (*lag*) among all pairs of points and apply to their values the equation 1.

Once obtained the experimental semivariogram, which basically is a plot showing how the semivariance of our dataset changes while increasing the separating distance between two points, we can get the best fitting curve called "theoretical semivariogram" (Figure 8). The latter gives us useful information and parameters as (Taylor et al., 2007):

- Nugget (c_0), which gives an estimation of the semivariance at a shorter distance than the smallest sampling distance (*lag 0)*.

- Sill ($c_0 + c_1$), which estimates the overall amount of variance in the data.
- Range (a) that estimates the distance or *lag* at which the spatial autocorrelation vanishes.

Starting from those parameters, it is possible to calculate the "Cambardella Index" (CI), which values show strong spatial correlation if lower than 25, moderate autocorrelation if in between 25 and 75, weak autocorrelation if higher than 75:

$$CI = \frac{c_0}{c_0 + c_1} \times 100 \qquad \text{(Equation 2)}$$

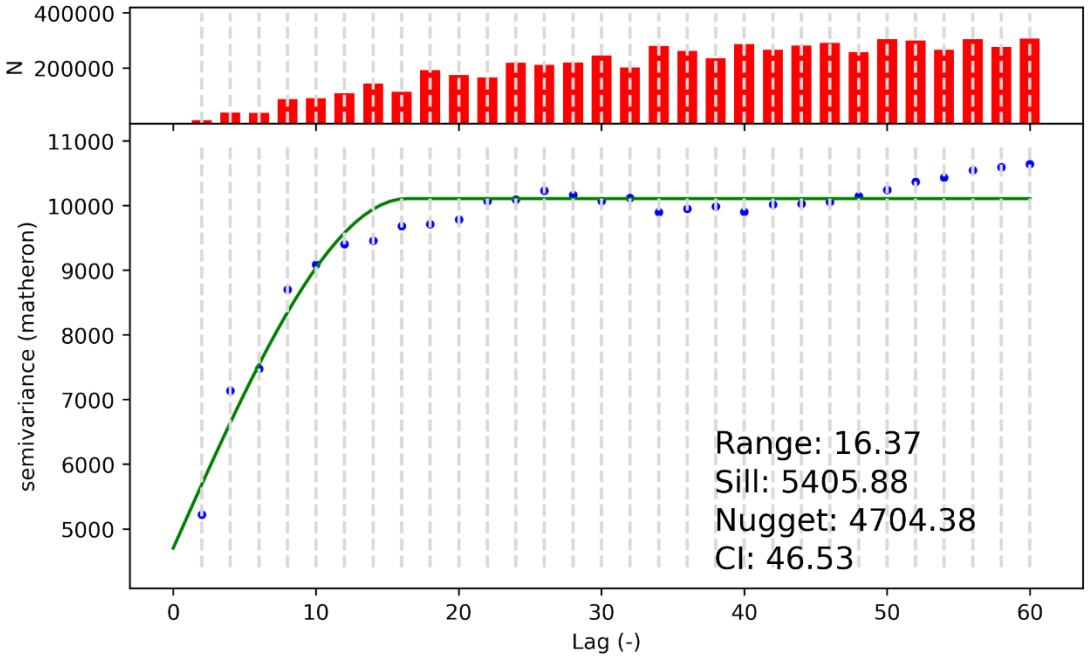


Figure 8. spherical semivariogram of the estimated flower cluster load in a Fuji orchard

1.7. THESIS OBJECTIVES

The main aim of this dissertation is to address all the listed “drawbacks” at paragraph 1.2 on processing drone imagery to get flowering maps starting from RGB image segmentation. the goal is to collect all the solutions into an easy-to-use Python program, which potentially if translated into a Graphical User Interface (GUI) could support apple growers to reach the farm sustainability.

Under this circumstance some requirements are needed:

- reducing image processing time,
- avoidance of expensive technology,
- high reliability,
- full automation of the processes,
- adaptability to multiple tree’s training systems,
- avoidance of the need for powerful computers,
- ease of use.

2. MATERIALS AND METHODS

2.1. DATA COLLECTION

The data were collected on the 10th of April 2021, on a 1.6 ha 'Fuji' (cv. Aztec) apple orchard located in the Po valley (44.7651278 N – 11.7587535 E) and characterised by 2004 as planting year, tree spacing of 3.3 m inter – row and 1.0 m intra – row whit super spindle training system.

The collected data refer both to georeferenced flower cluster load of hundred-one apple trees that were manually sampled within the study area, and to the top-side view aerial images of the orchard got by a Parrot ANAFI Thermal's (Parrot Drones SAS - www.parrot.com) RGB camera. The day of the data collection the apple trees were at "full bloom" phenological stage (Figure 9), presenting a bit of scalarity with flower clusters at "red buds" stage (BBCH 60 - 65). The overall cluster load per tree was calculated as the sum of all the clusters in between BBCH 60-65 on both the tree's walls. The georeferencing of manually sampled data consisted in creating a database to associate to each cluster load the GPS coordinate (WGS84 datum) of its respective tree.

To what concerns the aerial survey, it was performed by exploiting the 16MP RGB camera (frame: 4608x3456 / 4:3/ HFOV 75.5°) of the Anafi thermal ("Product sheet," n.d.) which has an optical unit characterised by aspheric lenses and rectilinear resolution to minimize and uniform distortions at image edges, while maximizing the performances at the centre of the image. The desired average flight elevation or 'ground distance' was set to 24m in order to get a spatial resolution (Ground Sampling Distance – GSD) of 1.02 cm/pixel, but weather conditions obstructed this reaching.

Figure 9. 10th of April 2021, cv. Fuji Aztec phenological stage (BBCH 60-65)

2.2. DATA ANALYSIS AND PROCESSING

Data analysis mainly consisted into image manipulations within a workflow that starts from metadata extraction and ends with flower detection by binary segmentation. Following that, geospatial analysis was performed to produce vector shapefiles, raster maps and semivariogram models able to describe the whole orchard flowering variability.

2.2.1. METADATA EXTRACTION AND GPS CORRECTION

The first step of image manipulation is metadata extraction for useful information which allow frame's direct georeferencing. Thus, referring to: GPS location (longitude, latitude, altitude), drone and camera's yaw, pitch and roll (Figure 10).

Following the metadata extraction, the drone's Exif GPS location was corrected by altitude (AGL), camera yaw (ψ), pitch and roll (Figure 11) in order to calculate at ground level, the Euclidean distance in meters (|d|m) between the drone ground-projection and the real GPS location of the centre of the image.

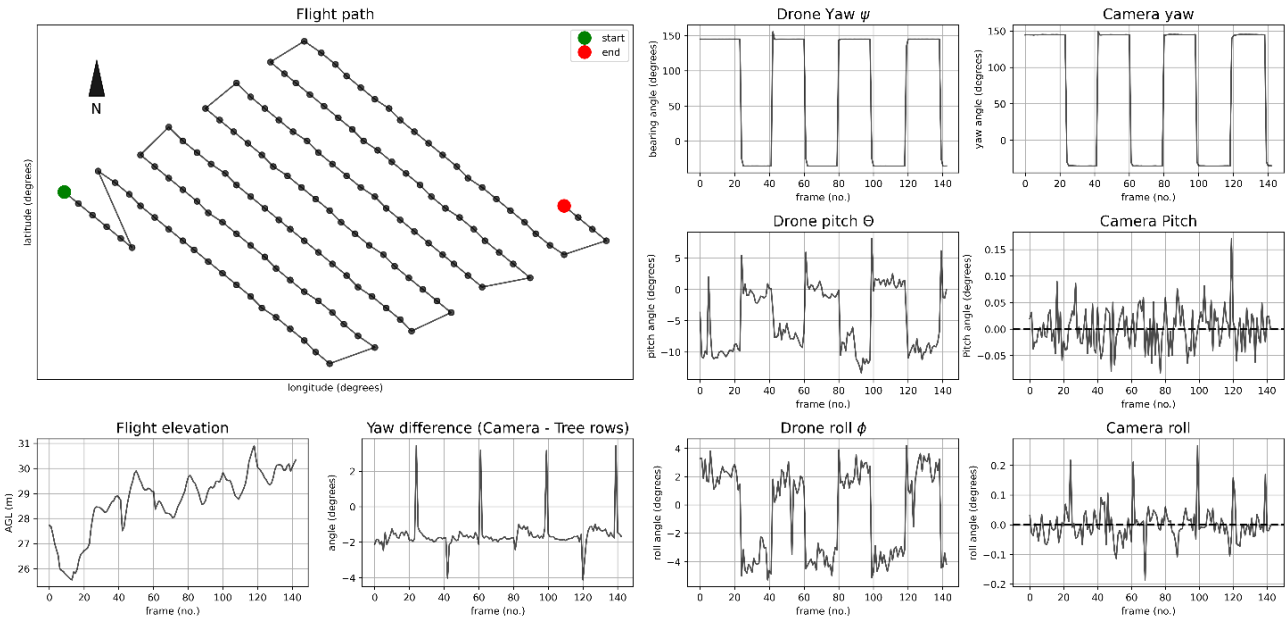
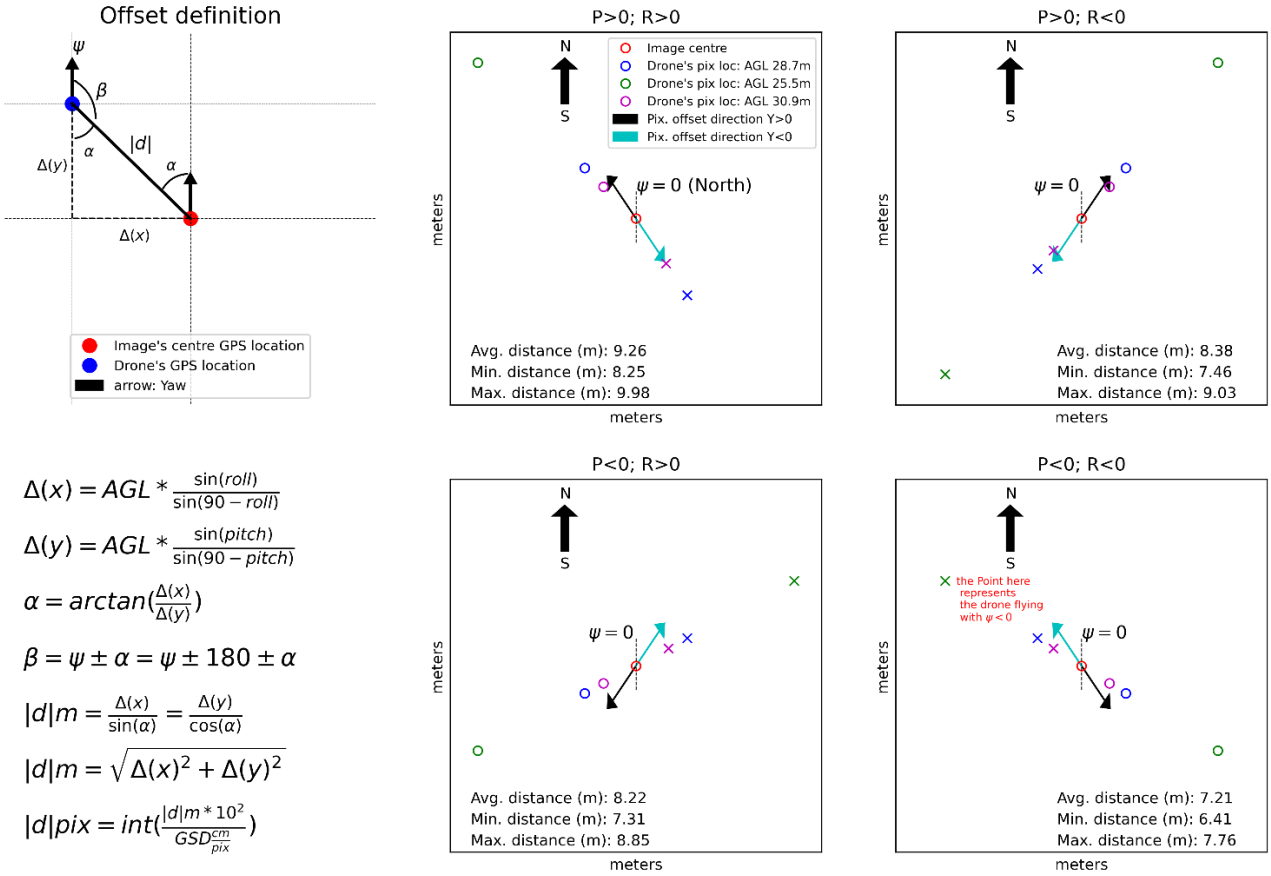


Figure 10. Photograms' metadata analysis: a) flight path from GPS coordinates, b) Drone yaw, c) camera yaw, d) drone pitch, e) camera yaw, f) drone roll, g) camera roll, h) flight elevation and i) yaw difference for image georeferencing correction.



$$\Delta(x) = AGL * \frac{\sin(roll)}{\sin(90 - roll)}$$

$$\Delta(y) = AGL * \frac{\sin(pitch)}{\sin(90 - pitch)}$$

$$\alpha = arctan(\frac{\Delta(x)}{\Delta(y)})$$

$$\beta = \psi \pm \alpha = \psi \pm 180 \pm \alpha$$

$$|d|m = \frac{\Delta(x)}{\sin(\alpha)} = \frac{\Delta(y)}{\cos(\alpha)}$$

$$|d|m = \sqrt{\Delta(x)^2 + \Delta(y)^2}$$

$$|d|pix = int(\frac{|d|m * 10^2}{GSD\frac{cm}{pix}})$$

Figure 11. correction of image centre GPS coordinate

After it, the metadata extracted longitude and latitude with the calculated offset were introduced inside the '*Haversine formula*' (Equation 3) to get the new latitude and longitude (Equation 4, 5) referred to the image centre.

$$hav(\theta) = sin^2\left(\frac{\theta}{2}\right) = hav(\varphi_2 - \varphi_1) + cos(\varphi_1)cos(\varphi_2)hav(\lambda_2 - \lambda_1) \quad \text{(Equation 3)}$$

$$\varphi_2 = \sin^{-1}\left(\sin\left(\frac{\pi \cdot \varphi_1}{180}\right) \cdot \cos\left(\frac{d}{R}\right) + \cos\left(\frac{\pi \cdot \varphi_1}{180}\right) \cdot \sin\left(\frac{d}{R}\right) \cdot \cos\left(\frac{\pi \cdot \beta}{180}\right)\right) \cdot \frac{180}{\pi} \quad \text{(Equation 4)}$$

$$\lambda_2 = \left(\left(\frac{\pi \cdot \lambda_1}{180}\right) + \tan^{-1}\left(\frac{\sin\left(\frac{\pi \cdot \beta}{180}\right) \cdot \sin\left(\frac{d}{R}\right) \cdot \cos\left(\frac{\pi \cdot \varphi_1}{180}\right)}{\cos\left(\frac{d}{R}\right) - \sin\left(\frac{\pi \cdot \varphi_1}{180}\right) \cdot \sin\left(\frac{\pi \cdot \varphi_2}{180}\right)}\right)\right) \cdot \frac{180}{\pi} \quad \text{(Equation 5)}$$

where: φ_1, φ_2 are latitude of point 1 and point 2; λ_1, λ_2 are longitude of point 1 and point 2; R is the Earth radius; d is the distance in km between point 1 and 2; β is the bearing angle between point 1 toward point 2 as showed in Figure 11.

2.2.2. IMAGE CLIPPING

After applying the GPS correction, there is image clipping, designed for reducing computing time and adaptation to different tree's training systems. It consists into removing from each frame the overlapped areas (Figure 12) in order to make them uniquely able to supply high resolution flowering information of a limited orchard area. Making it simpler, after the image clipping phase, there are not two frames covering the same area of interest (AOI). This is essential considering that reducing the number of pixels to be analysed in forward steps is a way to reduce computing time and also to increase the efficiency of the image analysis: each pixel gives an information that no other pixel in the dataset may give. To what concerns the adaptation to different training systems, the latter were distinguished into:

- 2 – dimensional training systems (2D), as the newly introduced multi-leader, which require a tree's side – view to properly quantify the cluster load,
- 3 – dimensional training systems (3D), as the super – spindle, for which a top – view of the tree is enough for cluster load quantification.

To make this distinction effective in the clipping process, the frame distortions were considered and clipping focused on image edges is performed to have a tree's side – view, while clipping focused on the image central, less distorted zone for a tree's top – side view (Figure 13).

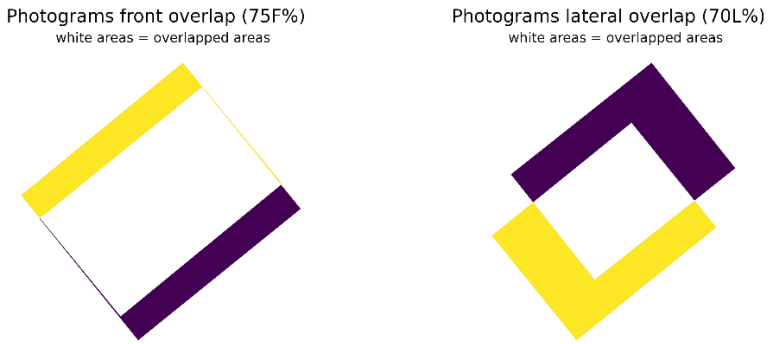


Figure 12. Aerial photograms overlapped areas: frontal and lateral.

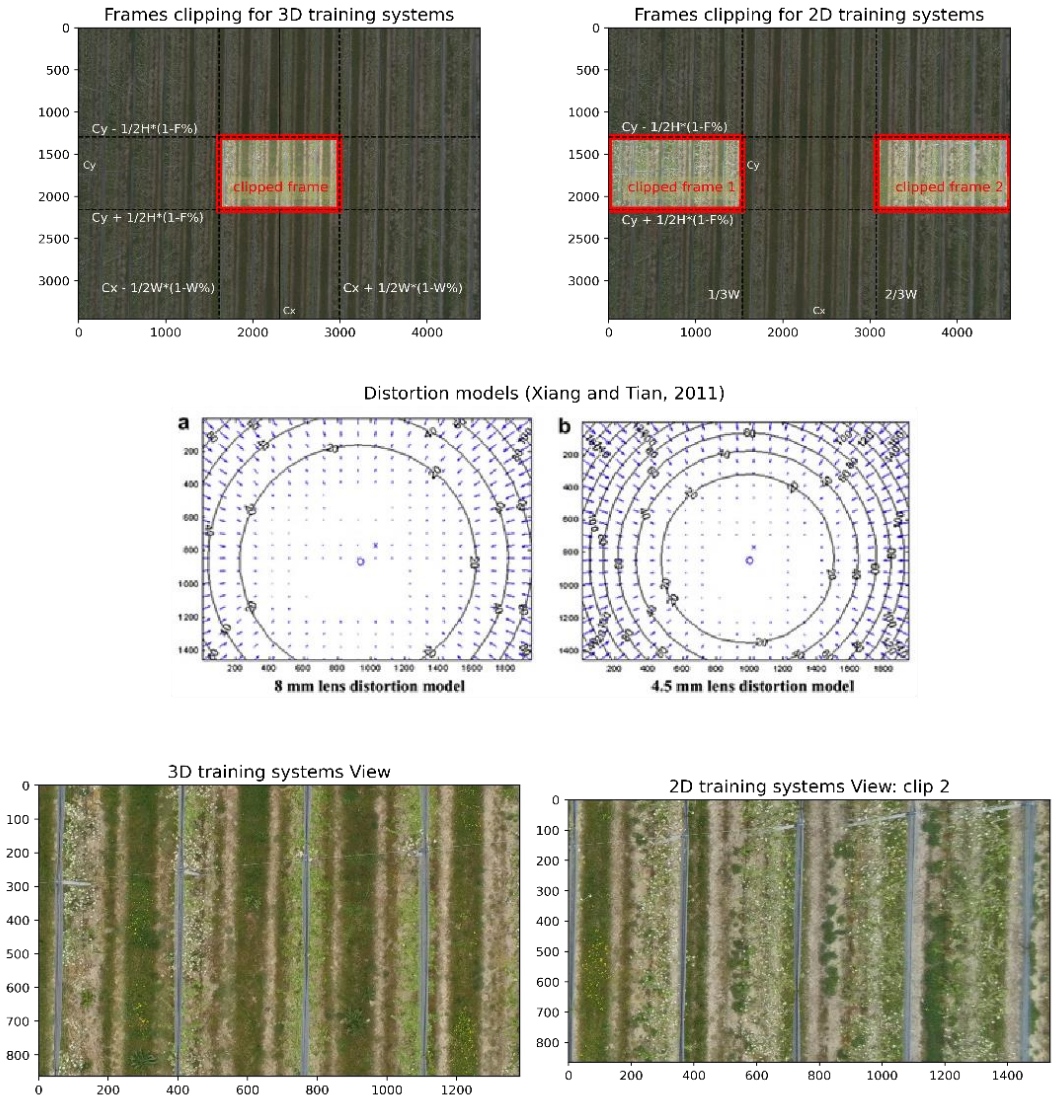


Figure 13. Image clipping for different training systems and reduced processing time (image size is reduced by 92 - 93%)

2.2.3. DIRECT GEOREFERENCING OF CLIPPED FRAMES

The direct georeferencing process is split into two complementary subprocess:

1. calculation of the clipped frame's corners GPS locations as Ground Control Points (GCPs),
2. clipped frame's metadata and file format transformation from jpeg to GTiff.

The first, consists into setting a within image coordinate reference system according to the chosen clipping strategy, which allows to calculate from the image centre geographic coordinate, based on the *Haversine equation* (Equation 3), the GPS location of the centre of the clipped frame, and then

from it the four corners' (UL - upper left, UR – upper right, BR – bottom right, BL – bottom left) GPS coordinates as GCPs for the direct georeferencing. In those calculations the camera yaw is considered (Figure 14).

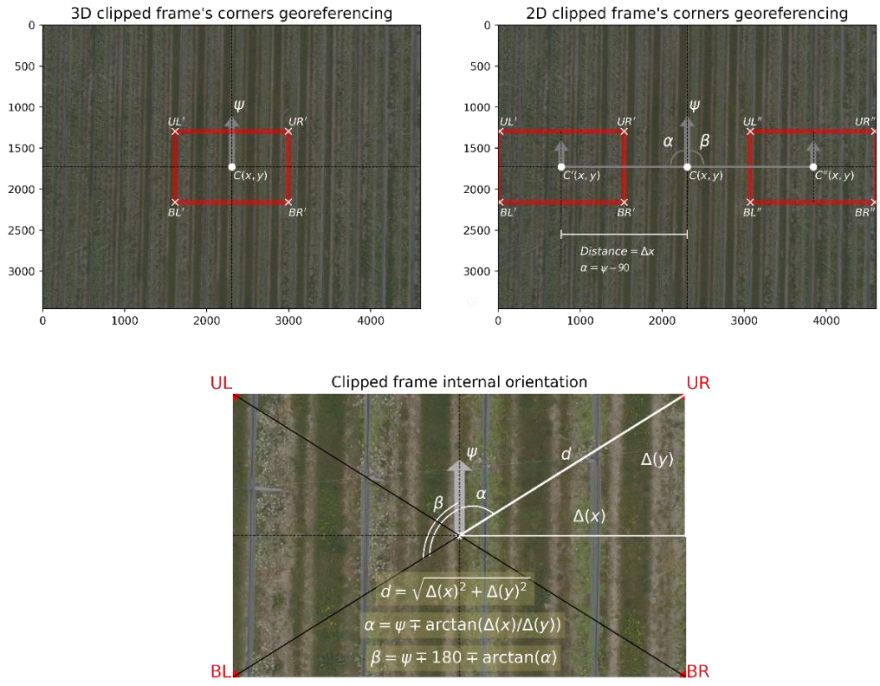


Figure 14. Within image coordinates reference system for GCPs calculation

The second subprocess (Figure 15) consists into assigning to image's centre and corners pixels the GCPs. For doing that, image format changes and metadata manipulation are needed. Firstly, the clipped frame is converted from .jpeg to .tiff file, secondly the .tiff file is assigned with the proper geographical Coordinates Reference System (CRS), which in this case is the WGS84 Datum (EPSG: 4326), then the GCPs are used to assign to the pixels of the four corners and the image centre the corresponding geographical location. Thirdly, the tiff raster with GCPs is warped in order to georeference the whole raster and make its file format effective as GeoTiff. The last step is geotransform function exploitation to rotate the raster according to the difference between the camera yaw and orchard tree rows' bearing, which allows to further warp the raster and get more precise georeferencing.

1. Clipped image

```
Driver: JPEG/JPEG JFIF
Files: C:\Users\mirko\Desktop\P5310948_clip.jpg
Size is 1372, 784
Image Structure Metadata:
  COMPRESSION=JPEG
  INTERLEAVE=PIXEL
  SOURCE_COLOR_SPACE=YCbCr
Corner Coordinates:
Upper Left (    0.0,    0.0)
Lower Left (    0.0,  784.0)
Upper Right ( 1372.0,    0.0)
Lower Right ( 1372.0,  784.0)
Center      (  686.0,  392.0)
Band 1 Block=1372x1 Type=Byte, ColorInterp=Red
  Overviews: 686x392, 343x196, 172x98
  Image Structure Metadata:
    COMPRESSION=JPEG
Band 2 Block=1372x1 Type=Byte, ColorInterp=Green
  Overviews: 686x392, 343x196, 172x98
  Image Structure Metadata:
    COMPRESSION=JPEG
Band 3 Block=1372x1 Type=Byte, ColorInterp=Blue
  Overviews: 686x392, 343x196, 172x98
  Image Structure Metadata:
    COMPRESSION=JPEG
```

2. Tiff image

```
Driver: GTiff/GeoTIFF
Files: C:\Users\mirko\Desktop\P5310948_clip_tiff.tiff
Size is 1372, 784
Image Structure Metadata:
  INTERLEAVE=PIXEL
Corner Coordinates:
Upper Left (    0.0,    0.0)
Lower Left (    0.0,  784.0)
Upper Right ( 1372.0,    0.0)
Lower Right ( 1372.0,  784.0)
Center      (  686.0,  392.0)
Band 1 Block=1372x1 Type=Byte, ColorInterp=Red
Band 2 Block=1372x1 Type=Byte, ColorInterp=Green
Band 3 Block=1372x1 Type=Byte, ColorInterp=Blue
```

3. GCP Tiff image

```
Driver: GTiff/GeoTIFF
Files: C:\Users\mirko\Desktop\P5310948_GCP.tiff
Size is 1372, 784
GCP Projection =
GEOGCRS["WGS 84",
    DATUM["World Geodetic System 1984",
        ELLIPSOID["WGS 84",6378137,298.257223563,
            LENGTHUNIT["metre",1]]],
    PRIMEM["Greenwich",0,
        ANGLEUNIT["degree",0.0174532925199433]],
    CS[ellipsoidal,2],
        AXIS["geodetic latitude (Lat)",north,
            ORDER[1],
            ANGLEUNIT["degree",0.0174532925199433]],
        AXIS["geodetic longitude (Lon)",east,
            ORDER[2],
            ANGLEUNIT["degree",0.0174532925199433]],
    ID["EPSG",4326]]
Data axis to CRS axis mapping: 2,1
GCP[  0]: Id=1, Info=
        (686,392) -> (11.7569802155076,44.7654797918478,0)
GCP[  1]: Id=2, Info=
        (0,0) -> (11.7570812656945,44.7654894773083,0)
GCP[  2]: Id=3, Info=
        (1372,0) -> (11.7569406659652,44.7654130633958,0)
GCP[  3]: Id=4, Info=
        (0,784) -> (11.7570197651414,44.7655465202862,0)
GCP[  4]: Id=5, Info=
        (1372,784) -> (11.7568791653546,44.7654701062983,0)
Metadata:
  AREA_OR_POINT=Area
Image Structure Metadata:
  INTERLEAVE=PIXEL
Corner Coordinates:
Upper Left (    0.0,    0.0)
Lower Left (    0.0,  784.0)
Upper Right ( 1372.0,    0.0)
Lower Right ( 1372.0,  784.0)
Center      (  686.0,  392.0)
Band 1 Block=1372x1 Type=Byte, ColorInterp=Red
Band 2 Block=1372x1 Type=Byte, ColorInterp=Green
Band 3 Block=1372x1 Type=Byte, ColorInterp=Blue
```

4. GeoTiff raster

```
Driver: GTiff/GeoTIFF
Files: C:\Users\mirko\Desktop\P5310948_Gtiff.tiff
Size is 1573, 1039
Coordinate System is:
GEOGCRS["WGS 84",
    DATUM["World Geodetic System 1984",
        ELLIPSOID["WGS 84",6378137,298.257223563,
            LENGTHUNIT["metre",1]]],
    PRIMEM["Greenwich",0,
        ANGLEUNIT["degree",0.0174532925199433]],
    CS[ellipsoidal,2],
        AXIS["geodetic latitude (Lat)",north,
            ORDER[1],
            ANGLEUNIT["degree",0.0174532925199433]],
        AXIS["geodetic longitude (Lon)",east,
            ORDER[2],
            ANGLEUNIT["degree",0.0174532925199433]],
    ID["EPSG",4326]]
Data axis to CRS axis mapping: 2,1
GeoTransform =
  11.7568791653627, 1.2839392796666621e-07, -4.738929596739831e-09
  44.76554652027252, 4.738929596739831e-09, -1.2839392796666621e-07
Metadata:
  AREA_OR_POINT=Area
Image Structure Metadata:
  INTERLEAVE=PIXEL
Corner Coordinates:
Upper Left (  11.7568792,  44.7655465) ( 11d45'24.76"E, 44d45'55.97"N)
Lower Left (  11.7568742,  44.7654131) ( 11d45'24.75"E, 44d45'55.49"N)
Upper Right (  11.7570811,  44.7655540) ( 11d45'25.49"E, 44d45'55.99"N)
Lower Right (  11.7570762,  44.7654206) ( 11d45'25.47"E, 44d45'55.51"N)
Center      (  11.7569777,  44.7654835) ( 11d45'25.12"E, 44d45'55.74"N)
Band 1 Block=1573x1 Type=Byte, ColorInterp=Red
Band 2 Block=1573x1 Type=Byte, ColorInterp=Green
Band 3 Block=1573x1 Type=Byte, ColorInterp=Blue
```

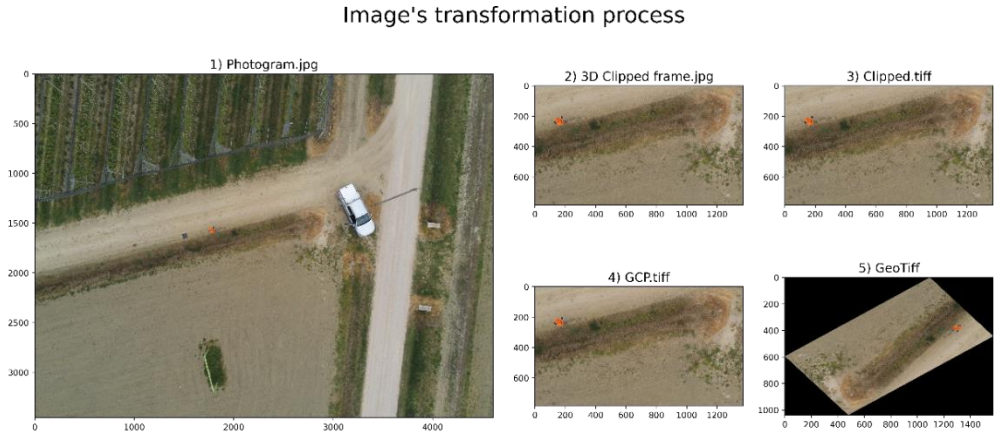


Figure 15. a) metadata transformation; b) image transformation and warp

2.2.4. GEOPROCESSING AT TREE LEVEL

To reach the tree level some geoprocessing was implemented in order to manipulate spatial data and exploit data fusion. Since modern technology as RTK GPS allows to plant trees precisely in space and produces as an output either a shapefile or comma separated value (.csv) file containing all the GPS locations of each tree in the orchard, the developed tool, allows the grower to use it as a database. It means that the proposed tool, for flower mapping, either exploits that database or generates it by itself starting from five given input points. Once the database containing trees' locations is available, within the geopandas framework, it is clipped according to the survey area. In this way more

processing time saving is reached and results are supplied to the user in advance, since the python program stops to quantify the flower load earlier, because of the avoidance of processing thousands of trees that fall outside the survey area. The latter is obtained by converting the GCPs of each clipped frame obtained over the georeferencing process, into a polygon shapefile and then all the polygons are merged into a single shapefile. Once the trees within the survey area have been identified (Figure 16-c), the further step is detecting for each frame which are the trees located in it. To reach this, the before mentioned frames' polygon shapefiles are exploited once again meaning that for each polygon, the list of trees within the survey area is clipped to identify the trees falling in the AOI (Figure 16-d). Once the trees have been detected, for each of them, according to tree rows bearing angle and planting distance, four GCPs are calculated and additional shapefiles representing the tree area are generated (Figure 16-e). The latter are then used to clip the georeferenced photogram (i.e., AOI) ang get a tree – specific image for the following flower detection phase (Figure 16-f).

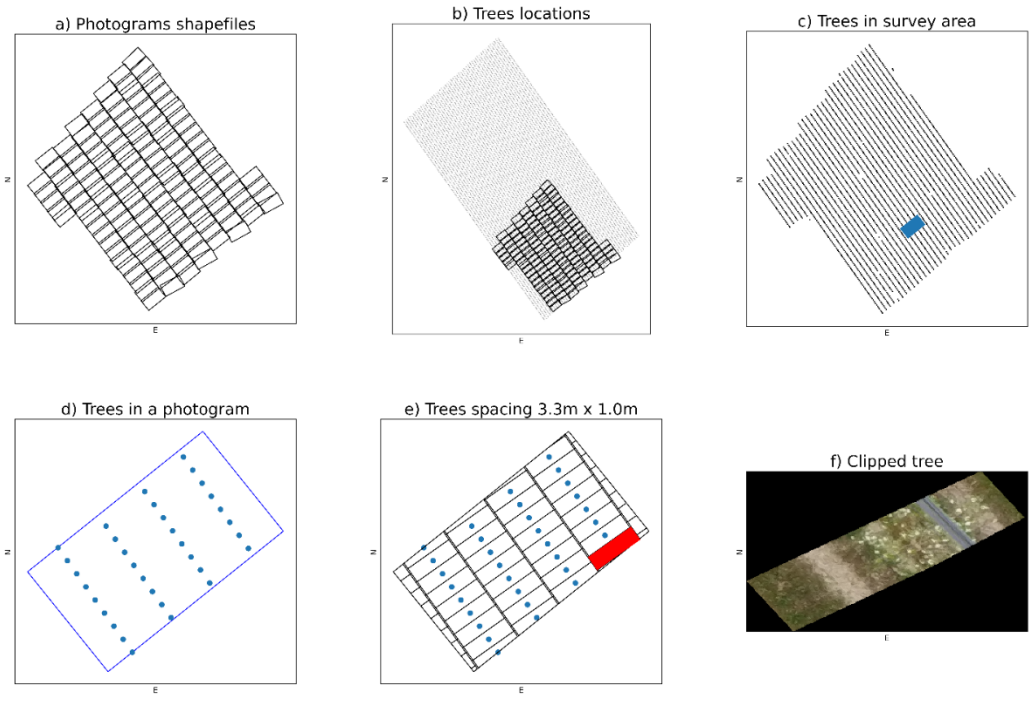


Figure 16. a) survey area; b) database of trees' locations with overlapped survey area; c) clipped database; d) trees in the photogram area; e) trees' bounding boxes with tree spacing dimensions; f) GeoTiff photogram clipped with the bounding box at tree level

2.2.5. FLOWER DETECTION

The detection of flower clusters is performed via image binary classification, in other words each pixel of the image is categorised by a defined query into "pixel containing flowers" and "pixel without flowers". To reduce at most the noise of the pixels not containing flowers, as those related to soil noise (Piani et al., 2021), the images are segmented into the HSV colour space (hue, saturation and brightness) instead of the RGB one. Additionally, the thresholds' values change dynamically according to the dimensions of the AOI, then if the image is tree – specific (Single Photogram Processing algorithm – SPP), the area is small and more homogeneous in terms of environmental and lightening conditions. As a consequence of it, the colour thresholding is calibrated to detect clusters at small scale when most of the noise is removed during the backward processes of image clipping and geoprocessing at tree level. Differently, if the AOI to process is larger, thus the whole clipped photogram (Photogram Clipping algorithm – PC), the environmental and lightening conditions are less uniform, and the colour thresholding is calibrated to handle with the noise and maximise flowers detection. As a direct result of the AOI's homogeneity, the flower clusters estimation is more effective at small scale rather than at larger one.

Table 1. thresholds values for flower detection

Thresholds	**3D SPP**		**2D SPP**		**PC**		
If: V max	**>= 242**	**< 242**	**>= 242**	**< 242**	**>= 230**	**>= 220**	**< 220**
H min	11	20	11	20	30	30	30
H max	179	179	179	179	179	179	179
S min	0	30	0	30	30	30	30
S max	255	255	255	255	255	255	255
V min	211	211	211	211	184	170	125
V max	255	255	255	255	255	255	255

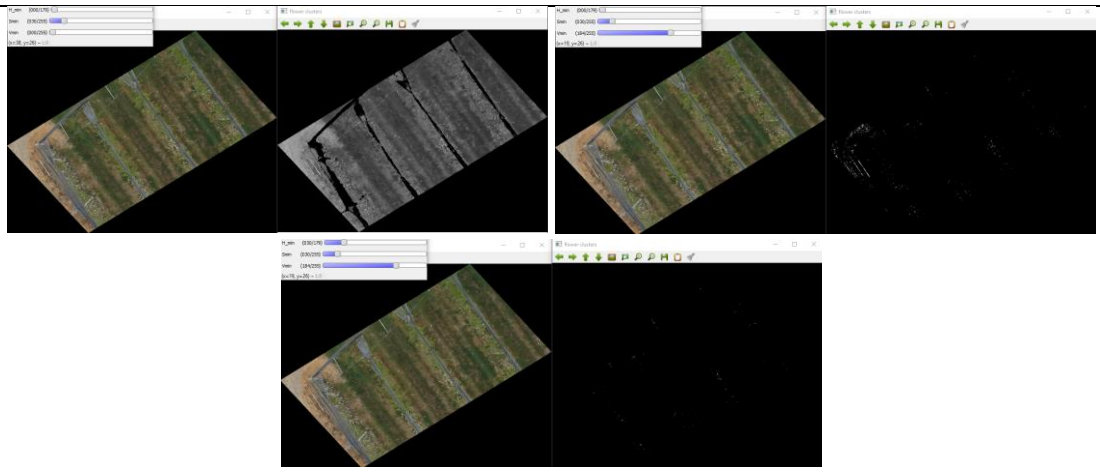
Referring to the thresholds' purposes (Figure 17):

- Hue minimum (H min) removes most of the soil noise,

- Saturation minimum (S min) removes poles and hail – nets noises,
- Value minimum (V min) removes vegetation and soil noises.

Following the images' binary classification according to the thresholds in Table 1, the flower clusters load in the AOI is estimated through the sum of all white pixels detected in it (i.e., "1 – value" pixels after query execution). That because after getting the sum of all the flower related pixels, it is converted to flower load by a calibration function specific for the adopted clipping process (2D at tree level, 3D at tree level, whole clipped photogram). The cited calibration functions are obtained by a regression of the independent variable "flower pixels" with the dependent variable "sampled flower clusters" both referred to the hundred – one sampled tree.

Photogram Clipping (PC) algorithm – HSV thresholds effects



Single Photogram Processing (SPP) algorithm – HSV thresholds effects

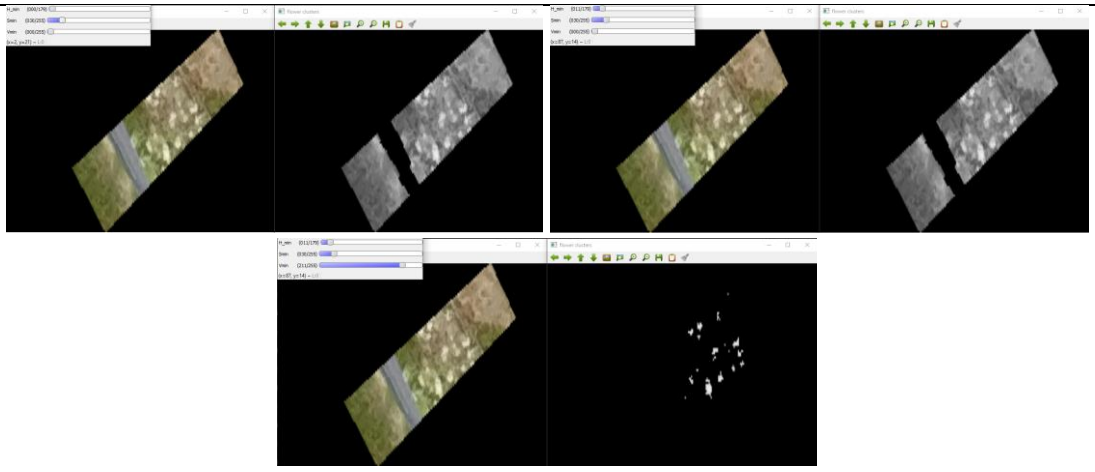


Figure 17. HSV thresholds' effects on flower detection

2.3. PROGRAM DEVELOPMENT

With the purpose of bringing full automation and ease of use to fruit growers, all the different steps described from paragraph 2.2.1 to 2.2.5 have been programmed in Python 3.10.1 [MSC v.1929 64 bit (AMD64)] and designed to reach both site-specific cluster load quantification and average area-specific cluster load quantification (PC) according to the degree of variability that the grower's machinery and employees are able to handle during mechanical, chemical or manual thinning. It follows that, the program which allows to reach site – specific 3D training systems cluster load quantification is called "Single Photogram Processing 3D" (3D SPP), while the one for site – specific 2D training systems quantification is "2D SPP". On the other hand, the program to get the average cluster load in the whole area of interest is called "Photogram Clipping" (PC). According to the degree of variability to handle, the algorithms have been programmed to give multiple information starting from the collection of simple user inputs (Figure 18) as:

- directory to the folder where the drone imagery is stored,
- survey resolution (GSD),
- frame overlaps percentages (frontal and lateral)
- tree spacing,
- database file containing trees' GPS coordinates or five field's reference points.

The main difference among the SPP algorithms and the PC one is that the latter following the image clipping directly proceeds with flower detection on the AOI, while SPP algorithms continue with geoprocessing up to tree level and site – specific flower detection. Another difference is that the SPP algorithms as outputs besides the maps showing the flowering spatial variability (Figure 19), give some semivariogram models (Spherical, Exponential and Gaussian) that the grower can use to manage spatial data within GIS software as QGIS and to perform manual sampling in the orchard during the production processes that follow blooming.

3D SPP and 2D SPP programs inputs

```
##########################################################
                        SURVEY DATA
##########################################################

Insert images' directory: C:\Users\mirko\Desktop\Program user inputs sample
Insert survey GSD (cm/pixel): 1.02

The recommended minimum frames' overlap should be:
 - 75% frontal and 60% side overlap in general cases.
 - 80% frontal and side overlap for agriculture fields  (****)
 - 85% frontal and side overlap for forests and dense vegetation.

Photograms frontal (H) overlap percentage: 75
Photograms side (W) overlap percentage: 70
##########################################################
                        FIELD DATA
##########################################################

Orchard training system (2D or 3D): 3D
In-row distance between trees (m): 1
Distance between tree rows (m): 3.3

Do you already have a .CSV file with trees' lat - long coordinates? (y/n): y

Are columns named and ordered as follows: lat | long ? (y/n) y
paste here the CSV file directory: C:\Users\mirko\Desktop\trees.csv
```

PC program inputs

```
##########################################################
                        SURVEY DATA
##########################################################

Insert images' directory: C:\Users\mirko\Desktop\Program user inputs sample
Insert survey GSD (cm/pixel): 1.02

The recommended minimum frames' overlap should be:
 - 75% frontal and 60% side overlap in general cases.
 - 80% frontal and side overlap for agriculture fields  (****)
 - 85% frontal and side overlap for forests and dense vegetation.

Photograms frontal (H) overlap percentage: 75
Photograms side (W) overlap percentage: 70



##########################################################
                        FIELD DATA
##########################################################

In-row distance between trees (m): 1
Distance between tree rows (m): 3.3
```

Figure 18. Programs' required user inputs

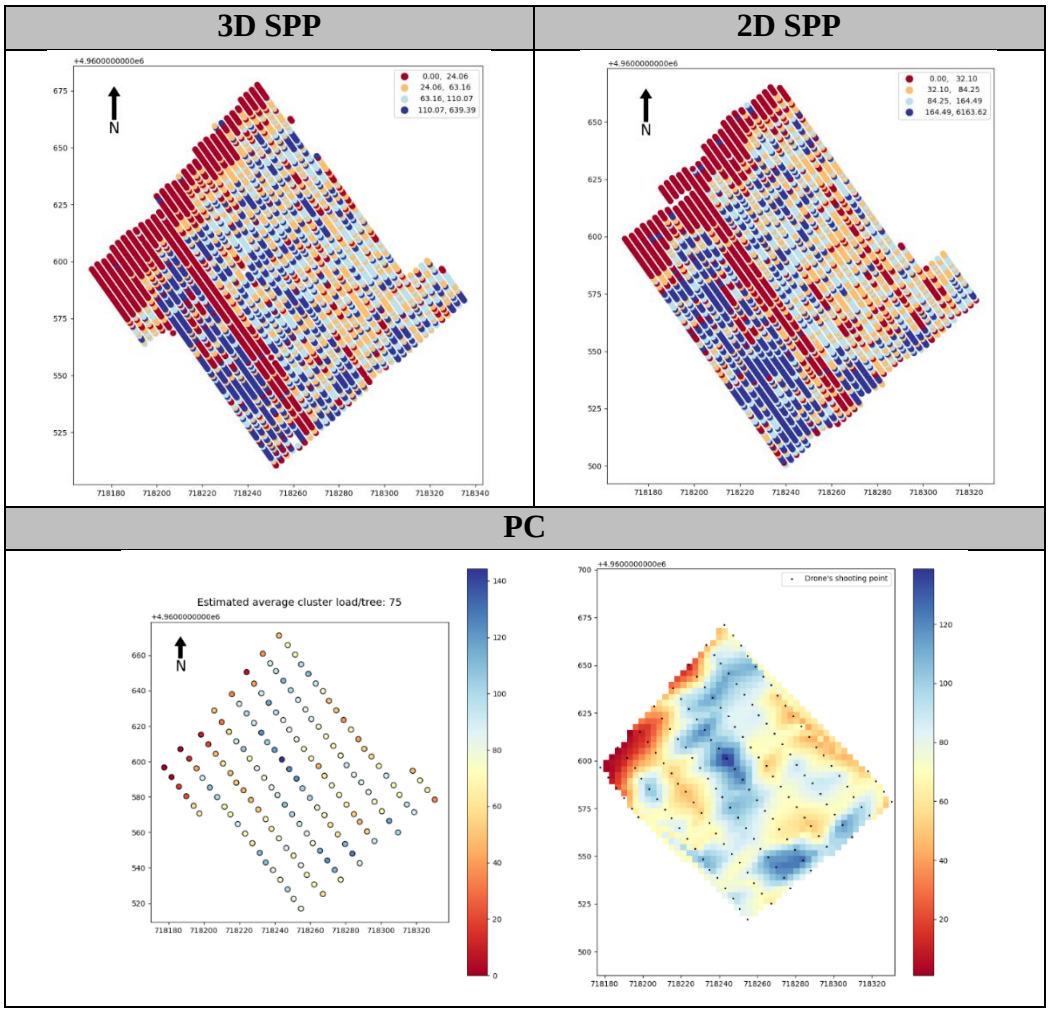


Figure 19. Programs' output map

2.3.1. ALGORITHMS CALIBRATION

Considering that the PC algorithm is unable to deep in space up to the tree level, to supply the user with an easily understandable information about cluster load quantifications or in other words the average amount of flower cluster per tree, a calibration process is needed.

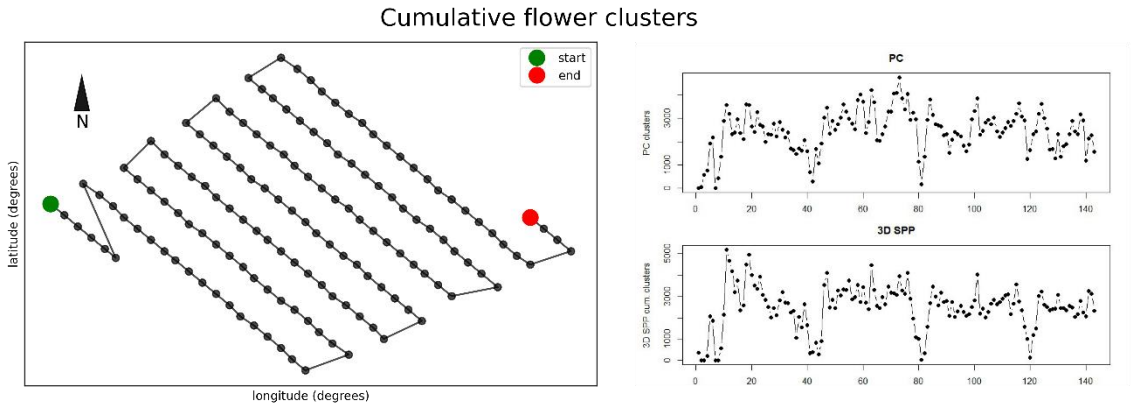


Figure 20. Cumulative SPP3D cluster load per photogram and PC one

This was performed by comparing (Figure 20) and doing a regression at photogram level, since the spatial resolution of the two algorithms is different, of the 3D SPP flower pixels values with the PC ones. After that the PC's load per photogram of flower – related pixels is converted with the regression model into cumulative 3D SPP pixels and following the conversion to 3D SPP cumulative cluster load per AOI. Then at last the average cluster load per tree is obtained by Equation 6.

$$Avg.cluster\ load\ per\ tree\ =\ SPP\ cumulative\ cluster\ load\ \cdot \frac{tree's\ area}{AOI\ area} \quad \text{(Equation 6)}$$

2.4. STATISTICAL ANALYSIS

The reliability of the algorithms in detecting flowering variability was evaluated at first by statistical analysis in *Rstudio* environment, thus in brief was analysed the variables correlation (*r*), the variables regression to find out through analysis of variance (ANOVA) and RMSE which is the best model to

convert the pixel load into cluster load per tree. Secondly a mapping analysis in *QGIS* environment was exploited to understand the models' spatial resolution and how they can be used to map the in – field flowering variability.

To exploit the correlation among the mentioned variables, and to get a regression model for 3D SPP and 2D SPP algorithms, it was used as independent variable (*x*) the flowers – related pixel load per tree and as dependent variable (y) the hundred sampled trees' cluster loads. The different tested regression models (linear, quadratic, cubic) were forced with intercept at axis origin in order to have as an output "no cluster load" for those trees showing no flower – related pixels. Otherwise, considering that the correlation coefficients are not equal to +1, the conversion to cluster load may give a completely wrong estimation given by the intercept different from zero. The effectiveness of the models in estimating the cluster load per tree was tested through the coefficient of determination (R^2), the root mean square error (RMSE) and the Analysis of variance (ANOVA).

3. RESULTS AND DISCUSSIONS

3.1. RELIABILITY

3.1.1. CORRELATION COEFFICIENT (r)

The correlation coefficient (r) of two variables is analysed to understand if somehow the variables are:

- positively correlated each other ($r > 0$),
- negatively correlated each other ($r < 0$),
- not correlated ($r = 0$).

For the analysis was exploited the maximum number of sampled trees located inside the georeferenced AOIs of both algorithms, then 97 observations for the 3D SPP algorithm and 50 for 2D SPP algorithm were considered.

The analysis pointed out that both 3D SPP and 2D SPP pixel loads are positively correlated with the sampled cluster loads with correlation coefficients respectively of 0.32 and 0.49. This suggests that a side – view of the tree allows to better quantify the flower cluster rather than a top – side view, but both of the views are effective. This correlation can be exploited in terms of conditional probability of dependent events (equation 7), since the within – tree flower presence can be determined in absence of phenomenon altering the flowers development by observing a part of the three. To make it simpler, when determining the amount of flower clusters per tree by manual sampling, it is usually quantified by counting the amount of inflorescence on a tree side and then multiplying it by a coefficient to get the whole tree flower clusters amount. That because it is feasible that after observing the presence of flowers on a side of the tree, the probability of having flowers on the other side of the tree depends on the observation. At the same way, observing flowers on the top of tree allows to understand that somehow even the bottom part of the tree has flowers randomly distributed.

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{favourable\ events}{possible\ events} \cdot \frac{1}{P(B)} \quad \text{(Equation 7)}$$

where: P(B) is the probability of counting some flowers on the side or top of the tree, P(A) is the probability of having flowers on the other side or bottom of the tree and P(A|B) is the probability of having the expression of event A knowing the realization of event B.

To make an example, the probability of either having or not flowers on a tree is 50%, P(B), so, if the counting flowers on a tree side is figuring out the presence of flowers, the possibility of having flowers on the other tree side is 1 out of 2 (50%), $P(A \cap B)$. Thus, if flowers on a side have/have not been counted, the probability of counting/not counting them on the other side is 100%, $P(A|B)$.

3.1.2. STATISTICAL ANALYSIS

For what concerns the coefficient of determination, it came out that increasing the degree of complexity of the model, then switching from a linear model to a cubic one, for both algorithms the phenomenon explanation increases from R^2 equal to 0.81 to 0.85 for 3D SPP and from 0.76 to 0.81 for 2D SPP. To summarise, all the models gave a degree of explanation higher than the ones proposed in the scientific literature which adopted image segmentation as quantification method (Piani et al., 2021, R^2 0.71). The ANOVA analysis of the models highlights that the increase in the phenomenon explanation while adding complexity is described by the significance given by the added coefficients modelling the variance of the dependent variable on the change in the measured independent variable as showed in Table 2.

Finally, the analysis of the RMSE pointed out that event if adding complexity to the models results in an increase in the explanation of the phenomenon, the errors in giving the right estimation get bigger as well (Figure 21). In that case, the use of linear regression models is assumed to be the one that supplies the most precise flower load estimation for field applications.

*Table 2. ANOVA analysis of the proposed models. Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1*

	3D SPP **x: flower pixels** **y: sampled clusters**	2D SPP **x: flower pixels** **y: sampled clusters**
Lin.	Analysis of Variance Table Response: y Df Sum Sq Mean Sq F value Pr (>F) x 1 671657 671657 403.2 < 2.2e-16 *** Resid. 96 159918 1666	Analysis of Variance Table Response: y Df Sum Sq Mean Sq F value Pr (>F) x 1 368067 368067 162.44 < 2.2e-16 *** Resid. 49 111027 2266
Quad.	Analysis of Variance Table Response: y Df Sum Sq Mean Sq F value Pr (>F) x 1 671657 671657 462.496 < 2.2e-16 *** x^2 1 21955 21955 15.118 0.0001871 *** Resid. 95 137963 1452	Analysis of Variance Table Response: y Df Sum Sq Mean Sq F value Pr (>F) x 1 368067 368067 186.35 < 2.2e-16 *** x^2 1 16218 16218 8.2108 0.006163 ** Resid. 48 137963 1452
Cubic	Analysis of Variance Table Response: y Df Sum Sq Mean Sq F value Pr (>F) x 1 671657 671657 519.596 < 2.2e-16 *** x^2 1 21955 21955 16.984 0.0001871 *** x^3 1 16454 16454 12.729 0.0005689 *** Resid. 94 121509 1293	Analysis of Variance Table Response: y Df Sum Sq Mean Sq F value Pr (>F) x 1 368067 368067 201.948 < 2.2e-16 *** x^2 1 16218 16218 8.8983 0.004515 ** x^3 1 16454 16454 5.0188 0.029840 * Resid. 47 85662 1823

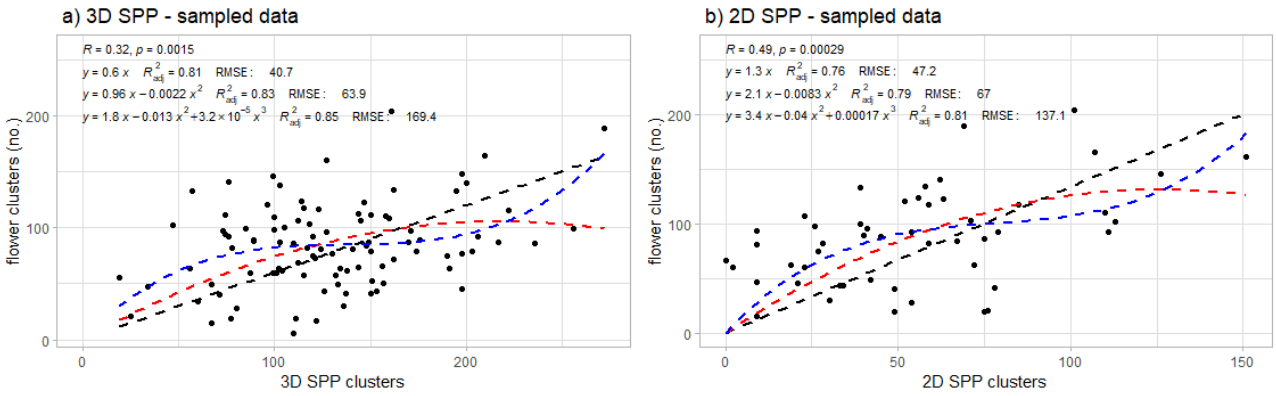


Figure 21. regression models (linear, quadratic, cubic) for both SPP algorithms

As mentioned in paragraph 2.3.1, the calibration function to convert the cumulative flower pixels detected by the PC algorithm into average cluster load per tree in the AOI, was evaluated in the same way as described for both the SPP algorithms, and similarly to them, the best fitting model to convert

PC's flower pixels into 3D SPP ones is the linear model (R^2 0.94, RMSE 1828.6, p < 2.2e-16) as showed in Figure 22.

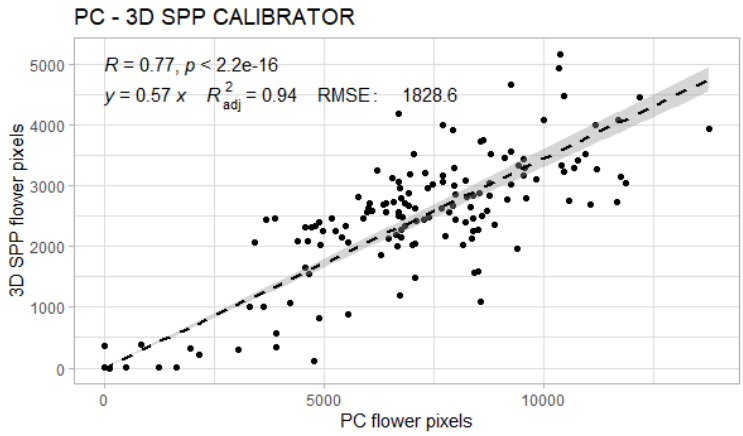


Figure 22. model to convert PC flower - related pixels to SPP flower pixels

The scatterplot shows clearly as increasing the size of the AOI (from tree – specific to whole photogram), the noise in colour thresholding becomes more evident leading to an overestimation of the flower load of about 40%. However, it is possible to adopt the whole photogram analysis to get a flowering trend which is quite precise in explaining the mean local variation.

To sum up, the dissertation confirmed that using cheap RGB technology is an effective way to map flower load over the orchard area. Additionally, this is effective since applying the SPP and PC algorithms at field level, it is possible to estimate loads instead of densities of flowers as other algorithms as the M5 one does (Piani et al., 2021), thus increasing the user potential perceived utility and data understanding.

3.1.3. MAPPING ANALYSIS OF SPATIAL RESOLUTION

To understand the spatial resolution of the programs, different strategies were adopted: the first one consisted into extracting from the databases supplied by both the SPP and PC algorithms the flower loads related to one of the tree rows that were manually sampled, and then by plotting for each

tree the estimated values with their respective sampled data, it was possible to notice for the SPP algorithms an in – row detection offset (Figure 23). The offset is originated by errors in the spatial orientation of the clipped photogram over the direct georeferencing process, and it is estimated to be in between one or two in – row tree positions (Figure 24).

To sum up, to what concerns the SPP algorithms, the spatial resolution concern the tree position (i.e., in this case a meter), while the error in the GPS location of the cluster load is in between 1 – 2 metres. Switching to the PC algorithm, from the same comparison, it is possible to notice that the averaging process removes the offset, and the spatial resolution is given by the photogram area, thus in this case 8.8m x 13.3m.

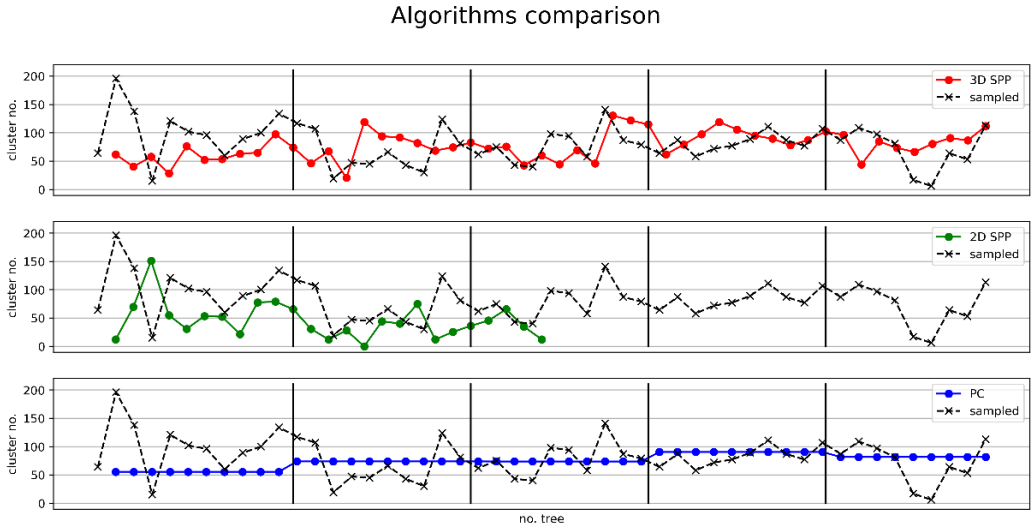


Figure 23. in row variability detection

Figure 24. differences in tree positioning given by the georeferencing method adopted

Once the errors embedded in the systems were detected, to evaluate their effect in producing prescription maps that could be used to organise the thinning procedures, the interpolation of all the discrete data into raster maps (Figure 25) was performed using a deterministic approach (*Inverse Distance Weighting* – IDW with $\alpha = 2$). From the visual analysis of the different raster maps it can be pointed out that to produce a prescription map, according to the degree of variability that the farmer can manage, both the SPP algorithms can substitute manual sampling if the target is a tree – based VRT thinning, while if the target agronomical intervention is a local VRT thinning, the PC algorithm may be the best one.

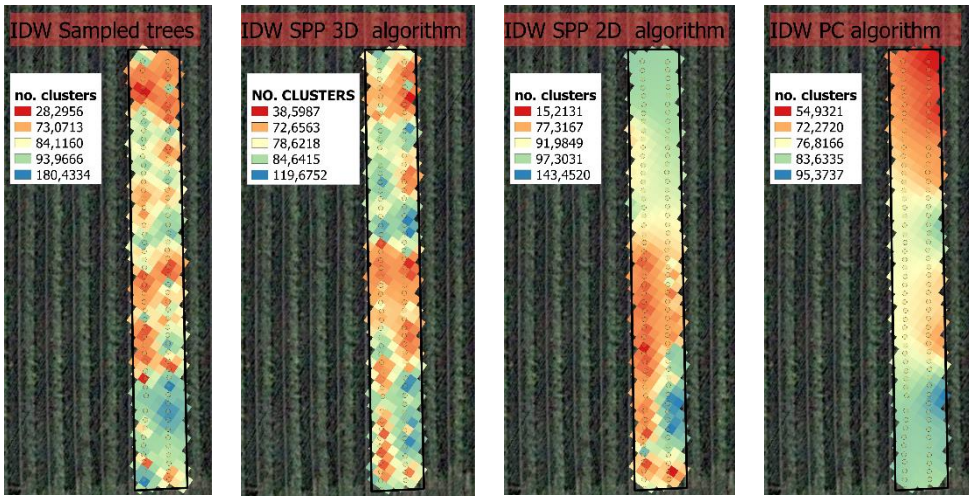


Figure 25. deterministic interpolations (IDW) over the sampled trees parcel

3.2. PROCESSING TIME

As reported by (Piani et al., 2021), in the process of mapping at field level the flower clusters density/load, the most time consuming phase is the creation of the orthomosaic. The proposed method for the direct georeferencing allows to get an orthomosaic – like raster of 1.6 ha orchard in 143 seconds, while producing an orthomosaic takes 6 hours (Figure 26).

Figure 26. georeferencing methods comparisons

Then, the time saving can be exploited to deepen flower detection up to tree – level. The geoprocessing process added with the flower detection requires 34 – 47 minutes for the 3D SPP algorithm and 40 – 47 minutes for the 2D SPP algorithm. Thus, in a comparison with the orthomosaic processing analysis (6h 25min) there is a net saving in processing time of about 89% for both SPPs algorithms.

Referring to the PC algorithm, which does not require image georeferencing and geoprocessing at tree level for the flower detection, it allows to get a geodatabase hosting average flower cluster loads information in 60 – 70 seconds, thus a net saving in processing time of 99.7%.

3.3. EASE OF USE AND PERCEIVED UTILITY

All the programs are coded in order to get simple inputs from the user related to the UAV survey and field characteristics as planting distance, training system and either a database storing trees' GPS locations or five reference points (Figure 27) from which to calculate trees position. Through data analysis, manipulation and transformation the value of the inputs data is increased, allowing the

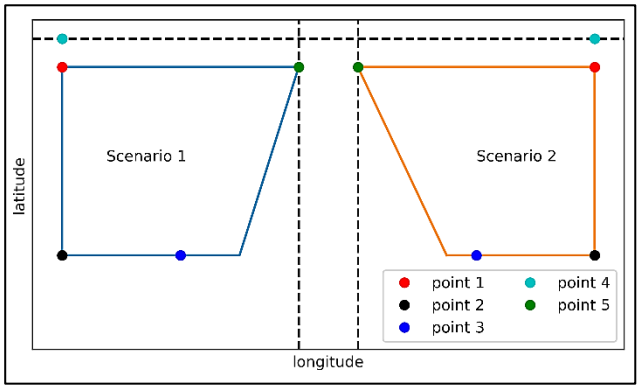


Figure 27. input reference points

user to get a final report containing a set of information related to orchard tree rows orientation (i.e., N-S, W-E, NW-SE, etc), orchard area, field GPS coordinates (longitude and latitude) and UTM zone/EPSG code that can be exploited for analysing data within GIS software. Additionally, within the report are contained also basics statistics about the flower load as the count of detected trees, average flower cluster load, quantiles, number, and percentage of trees without flowers that are useful for monitoring the biennial bearing disorder. The last section of the report contains the most common semivariogram models (exponential, gaussian, spherical) and their characteristics in term of range, sill, nugget, RMSE in order to allow the data analyst to choose the proper geostatistical model to interpolate data or even to manage manual sampling.

At the end of the algorithms run, a folder (Figure 28) containing the report, the maps showing the spatial variability among the orchard, and shapefiles in UTM .crs are automatically created. This was designed to allow the user to have a first look of the field condition by looking at the map, then getting more information from the report and if more is needed, the shapefiles can be directly imported within GIS software to perform the required geoprocessing without the need for layer's CRS conversion to UTM zone. The data about flower load are stored within a *.csv* database that can be used by the farmer to create a multiyear database or to store manually sampled data over the production period since it reports trees' GPS location and related flower load.

The data increased value has the purpose of enhancing the perceived utility of the farmer, since they give crucial information for the adoption of precision orchard management techniques.

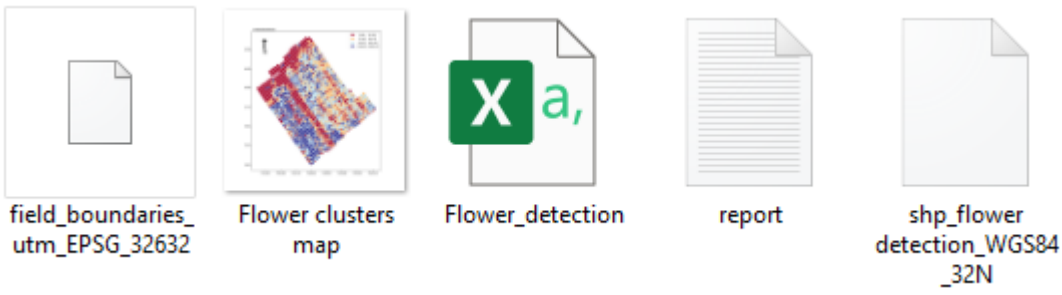


Figure 28. Python programs' output files

3.4. ADAPTABILITY TO FIELD CONDITIONS

The programs are developed to process top – view aerial images get by drone, thus the first limit for field application is the absence of closed hail nets. Otherwise, the presence of the net layer alters the colour thresholding for flower detection. Secondly, considering that most of the fruit growers have small size fields, and in most cases, those are heterogeneous in terms of training systems and cropped cultivars, allowing the farmer with a program which can handle different training systems (2D and 3D) to supply flowering information is a way to increase both the return of investment and the value of the collected images. That because, in case of multiple training systems, it is possible to perform just one flight to collect images, and then processing them with both the 3D and 2D algorithms. Later, inside a GIS software it is possible to clip and merge the output shapefiles given by the algorithms to describe the flower variability with the side – view for 2D training systems, and with top – view for 3D training systems.

3.5. FIELD OF APPLICATION

The developed programs are designed for field application characterised by high reliability and short processing time that should help the farmer in managing the thinning procedures and to optimize the crop load resulting in an increase in the overall orchard production. This is reached since the maps showing the spatial variability of the flower load may be used to produce prescription maps for thinning requiring machinery utilisation as mechanical and chemical thinning do (Figure 29).

Prescription maps can be produced even to manage the manual thinning which becomes essential to optimize production especially in organic farms, that because of the fruitlet competition for carbohydrates and the time required to thin a hectare by hand (300h/ha circa). It is clear as in the required timeframe the competition may slow the fruit growth of several decigrams per day according to the crop load. For this reason, by knowing zonal or tree – specific flower load, it is possible to

organize the thinning and the employees to act at first in those orchard's sectors where more fruitlets competition is expected to be realised and at last, to act in those areas having the less competition (Figure 30). This could result in homogenizing the fruit growth, hence fruit diameter and weight.

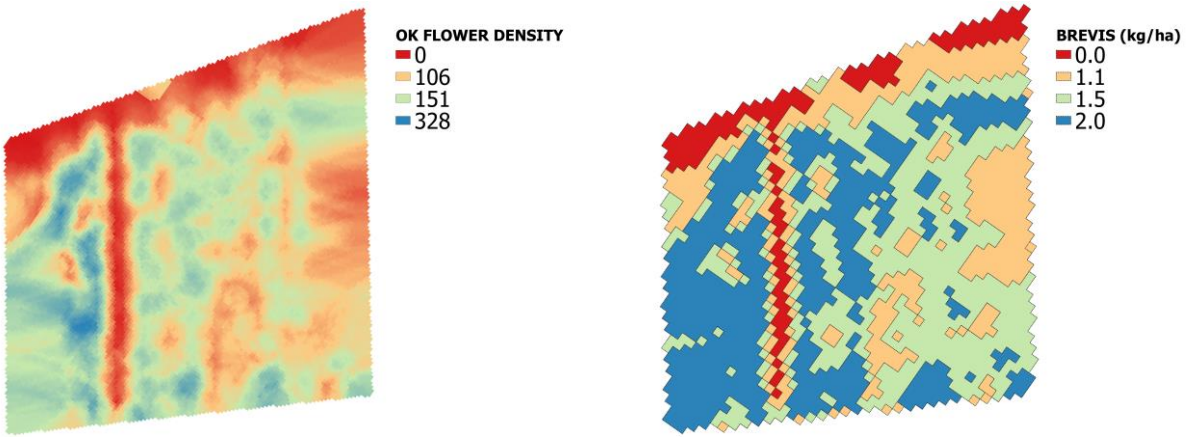


Figure 29. a) map of flower density, b) prescription map for chemical thinning

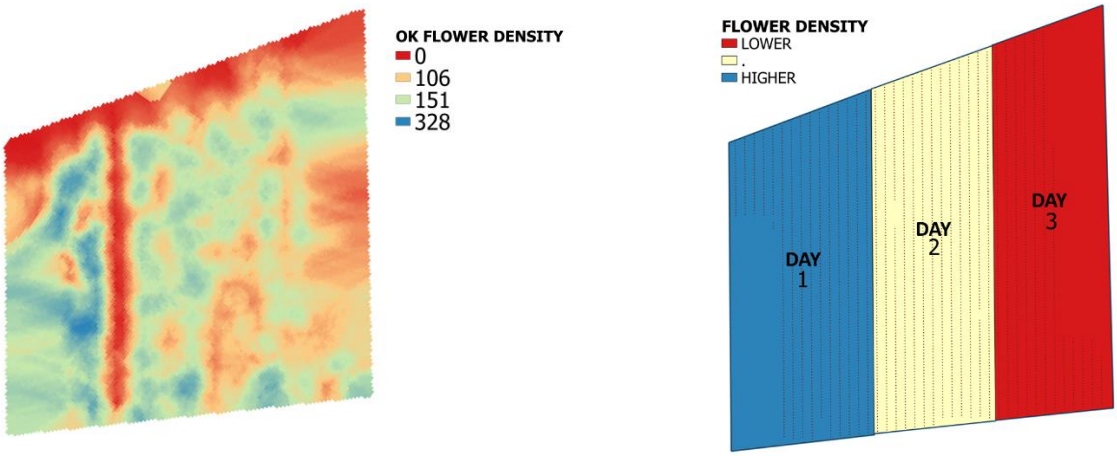


Figure 30. prescription map for manual thinning

BIBLIOGRAPHY

Basri, R., Islam, F., Shorif, S.B., Uddin, M.S., 2021. Robots and Drones in Agriculture—A Survey, in: Uddin, M.S., Bansal, J.C. (Eds.), Computer Vision and Machine Learning in Agriculture. Springer, Singapore, pp. 9–29. https://doi.org/10.1007/978-981-33-6424-0_2

Corradeghini, P., 2018. UN RILIEVO AEROFOTOGRAMMETRICO INIZIANDO DALLA SCALA DI RAPPRESENTAZIONE. 3dmetrica. URL https://3dmetrica.it/un-rilievo-fotogrammetrico-dalla-scala-di-rappresentazione/ (accessed 3.29.22).

Dalhaus, T., Schlenker, W., Blanke, M.M., Bravin, E., Finger, R., 2020. The Effects of Extreme Weather on Apple Quality. Sci. Rep. 10, 7919. https://doi.org/10.1038/s41598-020-64806-7

Daponte, P., Vito, L.D., Glielmo, L., Iannelli, L., Liuzza, D., Picariello, F., Silano, G., 2019. A review on the use of drones for precision agriculture. IOP Conf. Ser. Earth Environ. Sci. 275, 012022. https://doi.org/10.1088/1755-1315/275/1/012022

European Union, 2021, 2021. Ageing of Europe's farmers remains a major challenge in rural areas [WWW Document]. Eur. Comm. - Eur. Comm. URL https://ec.europa.eu/info/news/ageing-europes-farmers-remains-major-challenge-rural-areas-2021-apr-08_en (accessed 3.28.22).

Florin, M.J., McBratney, A.B., Whelan, B.M., 2010. Inverse Meta-modelling of Yield-Monitor Data for Estimating Soil-Available Water-Holding Capacities at a Farm Resolution of 10 m, in: Viscarra Rossel, R.A., McBratney, Alex B., Minasny, B. (Eds.), Proximal Soil Sensing, Progress in Soil Science. Springer Netherlands, Dordrecht, pp. 413–421. https://doi.org/10.1007/978-90-481-8859-8_35

Gautam, V., Sarkar, S., 2020. Smart Agriculture: The Age of Drones in Agriculture, in: Jain, K., Khoshelham, K., Zhu, X., Tiwari, A. (Eds.), Proceedings of UASG 2019. Springer International Publishing, Cham, pp. 415–424. https://doi.org/10.1007/978-3-030-37393-1_34

Lakso, A.N., Robinson, T.L., Pool, R.M., 1989. 17 - CANOPY MICROCLIMATE EFFECTS ON PATTERNS OF FRUITING AND FRUIT DEVELOPMENT IN APPLES AND GRAPES, in: Wright, C.J. (Ed.), Manipulation of Fruiting. Butterworth-Heinemann, pp. 263–274. https://doi.org/10.1016/B978-0-408-02608-6.50022-9

Liakos, V., Tagarakis, A., Aggelopoulou, K., Fountas, S., Nanos, G., Gemtos, T., 2017. In-season prediction of yield variability in an apple orchard. Eur. J. Hortic. Sci. 82, 251–259. https://doi.org/10.17660/eJHS.2017/82.5.5

Link, H., 2000. Significance of flower and fruit thinning on fruit quality. Plant Growth Regul. 31, 17–26. https://doi.org/10.1023/A:1006334110068

Manfrini, L. <1979>, 2009. Precision horticulture: application on apple orchards (Doctoral Thesis). Alma Mater Studiorum - Università di Bologna. https://doi.org/10.6092/unibo/amsdottorato/1627

Manfrini, L., Taylor, J., Grappadelli, L., 2009. Spatial Analysis of the Effect of Fruit Thinning on Apple Crop Load. Eur. J. Hortic. Sci. 74, 54–60.

Matheron, G., 1963. Principles of geostatistics. Econ. Geol. 58, 1246–1266. https://doi.org/10.2113/gsecongeo.58.8.1246

Moran, 1997. Opportunities and limitations for image-based remote sensing in precision crop management - ScienceDirect [WWW Document]. URL https://www-sciencedirect-com.ezproxy.unibo.it/science/article/pii/S003442579700045X?casa_token=Q4WVndvhzRcAAAAA:4FvvdvOg8ata7cUFtrP6tACMxGvt0yMZjI_nY5Vu6bXS32URvIXc8ZvnJFubZs5-oG3ClMal (accessed 3.29.22).

Naor, A., Stern, R., Flaishman, M., Gal, Y., Peres, M., 2006. Effects of post-harvest water stress on autumnal bloom and subsequent-season productivity in mid-season 'Spadona' pear. J. Hortic. Sci. Biotechnol. 81, 365–370. https://doi.org/10.1080/14620316.2006.11512074

Piani, M., Bortolotti, G., Manfrini, L., 2021. Apple orchard flower clusters density mapping by unmanned aerial vehicle RGB acquisitions, in: 2021 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor). Presented at the 2021 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor), pp. 92–96. https://doi.org/10.1109/MetroAgriFor52389.2021.9628565

Pierpaoli, E., Carli, G., Pignatti, E., Canavari, M., 2013. Drivers of Precision Agriculture Technologies Adoption: A Literature Review. Procedia Technol., 6th International Conference on Information and Communication Technologies in Agriculture, Food and Environment (HAICTA 2013) 8, 61–69. https://doi.org/10.1016/j.protcy.2013.11.010

Product sheet, n.d.

Ren, Y., Zhu, C., Xiao, S., 2018. Small Object Detection in Optical Remote Sensing Images via Modified Faster R-CNN. Appl. Sci. 8, 813. https://doi.org/10.3390/app8050813

Robinson, T.L., Francescatto, P., Lordan, J., 2021. Advances in precision crop load management of apple. Acta Hortic. 1314, 133–138. https://doi.org/10.17660/ActaHortic.2021.1314.18

Scalisi, A., McClymont, L., Underwood, J., Morton, P., Scheding, S., Goodwin, I., 2021. Reliability of a commercial platform for estimating flower cluster and fruit number, yield, tree geometry and light interception in apple trees under different rootstocks and row orientations. Comput. Electron. Agric. 191, 106519. https://doi.org/10.1016/j.compag.2021.106519

Štroner, M., Urban, R., Seidl, J., Reindl, T., Brouček, J., 2021. Photogrammetry Using UAV-Mounted GNSS RTK: Georeferencing Strategies without GCPs. Remote Sens. 13, 1336. https://doi.org/10.3390/rs13071336

Taylor, J.A., Praat, J.-P., Bollen, A.F., 2007. Spatial Variability of Kiwifruit Quality in Orchards and Its Implications for Sampling and Mapping. HortScience 42, 246–250. https://doi.org/10.21273/HORTSCI.42.2.246

Teodorescu, G., Moise, V., Cosac, A.C., 2016. Spatial Variation in Blooming and Yield in an Apple Orchard, in Romania. Ann. ”Valahia” Univ. Targoviste - Agric. 10, 1–6. https://doi.org/10.1515/agr-2016-0001

Tripicchio, P., Satler, M., Dabisias, G., Ruffaldi, E., Avizzano, C.A., 2015. Towards Smart Farming and Sustainable Agriculture with Drones, in: 2015 International Conference on Intelligent Environments. Presented at the 2015 International Conference on Intelligent Environments, pp. 140–143. https://doi.org/10.1109/IE.2015.29

van der Merwe, D., Burchfield, D.R., Witt, T.D., Price, K.P., Sharda, A., 2020. Chapter One - Drones in agriculture, in: Sparks, D.L. (Ed.), Advances in Agronomy. Academic Press, pp. 1–30. https://doi.org/10.1016/bs.agron.2020.03.001

Vanbrabant, Y., Delalieux, S., Tits, L., Pauly, K., Vandermaesen, J., Somers, B., 2020. Pear Flower Cluster Quantification Using RGB Drone Imagery. Agronomy 10, 407. https://doi.org/10.3390/agronomy10030407

Wang, X. (Annie), Tang, J., Whitty, M., 2020. Side-view apple flower mapping using edge-based fully convolutional networks for variable rate chemical thinning. Comput. Electron. Agric. 178, 105673. https://doi.org/10.1016/j.compag.2020.105673

Xiang, H., Tian, L., 2011. Method for automatic georeferencing aerial remote sensing (RS) images from an unmanned aerial vehicle (UAV) platform. Biosyst. Eng. 108, 104–113. https://doi.org/10.1016/j.biosystemseng.2010.11.003

Yang, B., Xu, Y., 2021. Applications of deep-learning approaches in horticultural research: a review. Hortic. Res. 8, 123. https://doi.org/10.1038/s41438-021-00560-9

Zhang, C., Huang, T., Zhao, Q., 2019. A New Model of RGB-D Camera Calibration Based on 3D Control Field. Sensors 19, 5082. https://doi.org/10.3390/s19235082

Zhao, J., Zhang, X., Gao, C., Qiu, X., Tian, Y., Zhu, Y., Cao, W., 2019. Rapid Mosaicking of Unmanned Aerial Vehicle (UAV) Images for Crop Growth Monitoring Using the SIFT Algorithm. Remote Sens. 11, 1226. https://doi.org/10.3390/rs11101226

Table of Figures

Tables