ISPRS Journal of Photogrammetry and Remote Sensing 182 (2021) 78–95

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

An optimized approach for generating dense thermal point clouds from UAV-imagery

Alfonso López *, Juan M. Jurado, Carlos J. Ogayar, Francisco R. Feito

Department of Computer Science, University of Jaén, Spain

ARTICLE INFO

Keywords:
UAV imagery
Thermal imagery
Image processing
Point cloud
Occlusion
GPU computing

ABSTRACT

Thermal infrared (TIR) images acquired from Unmanned Aircraft Vehicles (UAV) are gaining scientific interest in a wide variety of fields. However, the reconstruction of three-dimensional (3D) point clouds utilizing consumer-grade TIR images presents multiple drawbacks as a consequence of low-resolution and induced aberrations. Consequently, these problems may lead photogrammetric techniques, such as Structure from Motion (SfM), to generate poor results. This work proposes the use of RGB point clouds estimated from SfM as the input for building thermal point clouds. For that purpose, RGB and thermal imagery are registered using the Enhanced Correlation Coefficient (ECC) algorithm after removing acquisition errors, thus allowing us to project TIR images into an RGB point cloud. Furthermore, we consider several methods to provide accurate thermal values for each 3D point. First, the occlusion problem is solved through two different approaches, so that points that are not visible from a viewing angle do not erroneously receive values from foreground objects. Then, we propose a flexible method to aggregate multiple thermal values considering the dispersion from such aggregation to the image samples. Therefore, it minimizes error measurements. A naive classification algorithm is then applied to the thermal point clouds as a case study for evaluating the temperature of vegetation and ground points. As a result, our approach builds thermal point clouds with up to 798,69% more point density than results from other commercial solutions. Moreover, it minimizes the build time by using parallel computing for time-consuming tasks. Despite obtaining larger point clouds, we report up to 96,73% less processing time per 3D point.

1. Introduction

Infrared Radiation (IR) provides valuable information to detect and describe objects in a scene, while thermal cameras utilize passive sensors to measure the radiation in a small portion of the IR spectrum (Thermal InfraRed (TIR) band). Three spectral ranges are typically used for thermography: short-wave (0.9–1.7 μm), mid-wave (3–5 μm) and long-wave (8–14 μm) (Vollmer and Möllmann, 2017; Gade and Moeslund, 2014). Thermal remote sensing presents a wide range of applications, although they can differ significantly for the three conventional wavebands. For instance, thermography has been previously used for quality control and monitoring in industrial environments (Alfredo Osornio-Rios et al., 2019; Vollmer and Möllmann, 2017), as well as for medical analysis (Lõrinczy, 2017), building inspection (Jarzabek-Rychard et al., 2020; Kylili et al., 2014), agriculture and animal applications (McManus et al., 2016; Tsouros et al., 2019), geological monitoring (Grechi et al., 2021), fire detection (Gade and Moeslund, 2014), etc.

Regarding the surveying platform, TIR imaging was originally focused on satellite and manned vehicles due to their military use. In the last years, inexpensive and lightweight thermal cameras have spread their use to scientific and civilian environments (Vollmer and Möllmann, 2017; Sledz et al., 2018) and thus have found applications in other fields. However, the resolution of TIR images is low. For instance, 640 × 480 and 640 × 512 pixels are frequent resolutions for consumer-grade products. Consequently, the maximum height is restricted when small spatial resolutions are required, e.g. below 10 cm (Vollmer and Möllmann, 2017). As a result, the use of Unmanned Aerial Vehicles (UAV) provides an alternative for satellite remote sensing when surveying small areas, as their flight altitude is more flexible and thus allows to provide information with higher resolution (Ribeiro-Gomes et al., 2017). Furthermore, UAVs are more economically feasible and easier to operate than other aerial vehicles (Tsouros et al., 2019).

The acquisition of high-resolution images provides an opportunity to estimate three-dimensional (3D) models from the surveyed scene, such as point clouds or digital surface models (DSM). Data for 3D modeling can either be provided by laser sensors, such as LiDAR (Yandun Narvaez

* Corresponding author.
E-mail addresses: allopezr@ujaen.es (A. López), jjurado@ujaen.es (J.M. Jurado), cogayar@ujaen.es (C.J. Ogayar), ffeito@ujaen.es (F.R. Feito).

https://doi.org/10.1016/j.isprsjprs.2021.09.022
Received 11 May 2021; Received in revised form 20 September 2021; Accepted 29 September 2021
Available online 23 October 2021

et al., 2017), or derived from imagery processed by photogrammetry algorithms, for instance, Structure from Motion (SfM) (Jiang et al., 2020) and Multi-View Stereo (MVS) (Furukawa and Hernández, 2015). Previous studies show the benefits of three-dimensional models for assessing and monitoring a scene in a wide variety of fields, even for thermal imagery. Classification algorithms for 3D points clouds are frequent in Precision Agriculture (PA); (Comba et al., 2018) detects vineyards and vine-rows from 3D multispectral point clouds, while (Jurado-Rodríguez et al., 2020b) is focused on the detection of vineyard trunks from RGB point clouds. The characterization and evaluation of natural environments have also been extensively studied (Jurado-Rodríguez et al., 2020a; Webster et al., 2018). Furthermore, 3D modeling is being introduced to other research areas (Grechi et al., 2021).

However, photogrammetry algorithms rely on image data for key-points detection. Therefore, estimating 3D point clouds from low-resolution thermal imagery is a challenging task. First, TIR images suffer from stronger noise in comparison with RGB images (Sledz et al., 2018), as well as aberration-induced blurring causing the spreading of the object radiance. As opposed to ideal optical imaging, the radiation of an object field is observed by neighboring detector elements (Vollmer and Möllmann, 2017). In these conditions, the number of extracted tie points in TIR images is smaller, and thus the resulting 3D point clouds are much sparser and less accurate (Jarzabek-Rychard et al., 2020; Hoegner et al., 2016a; Ham and Golparvar-Fard, 2013a; Westfeld et al., 2015), although the outcome seems to be more stable when further calibration is considered (Ribeiro-Gomes et al., 2017). On the other hand, the optimization of camera orientation parameters and reduction of alignment errors can be achieved through the use of Ground Control Points (GCP) (Sanz-Ablanedo et al., 2018). Nevertheless, identifying GCPs on TIR images represents a challenge in fields with non-uniform materials (Javadnejad et al., 2020).

The fusion of multiple imagery resources is also recurrent in many thermography applications. Most works combine RGB point clouds with TIR imagery (Javadnejad et al., 2020; Jarzabek-Rychard et al., 2020; Grechi et al., 2021; Webster et al., 2018; Hou et al., 2021; Ham and Golparvar-Fard, 2013b), while multispectral imagery is seldom merged with previous data sources (Comba et al., 2019; Jurado-Rodríguez et al., 2020a). The most frequent workflow reconstructs RGB and thermal points clouds separately and then aligns both of them through the optimization algorithm of Iterative Closest Point (ICP) (Webster et al., 2018; Grechi et al., 2021; Ham and Golparvar-Fard, 2013b). Therefore, it is assumed that both point clouds are correctly estimated and present similar shapes. Few studies work in 2D space and project the results into 3D point clouds (Javadnejad et al., 2020; Hou et al., 2021; Hoegner et al., 2016b). (Hou et al., 2021) generates sparse thermal point clouds based solely on feature points, while (Javadnejad et al., 2020) produces point clouds with higher density through the fusion of RGB and TIR images. (Hoegner et al., 2016b) also evaluates the registration of RGB and TIR images through edge detection and the projection to an RGB point cloud. However, previous studies rely on external software and do not represent automatic solutions. As a consequence, the proposed methodologies are rarely optimized. Furthermore, conventional methods for image registration, such as edge detection, are not appropriate for blurred and noisy images.

Registration of TIR data and Mobile Laser Scanning (MLS) is also frequent in the literature, either including RGB images (Hoegner et al., 2018) or not (Zhu et al., 2021). These systems are geometrically calibrated through the 3D boresight and lever-arm transformations (Hoegner et al., 2018; Javadnejad et al., 2020) to describe the relative differences between the laser scanner system and the camera coordinate system. In addition, (Hoegner et al., 2018) distributes several control points on the acquired environment to calibrate the imaging sensor system, while (Zhu et al., 2021) utilizes feature extraction for TIR images and point clouds and tries to match pairs of key-points manually. Therefore, the described methodologies work under controlled environments and can be labeled as semi-automatic methods.

Moreover, ICP based works present more accurate results on their color distribution as MVS is an optimization problem constrained by a photo-consistency function (Furukawa and Hernández, 2015). Nevertheless, the inverse projection frequently ignores the occlusion problem, as occurs in (Javadnejad et al., 2020), where a 3D point can be projected into a viable pixel that, in fact, represents another object. In such a case, final colors are computed as an aggregation of feasible intensity values. (Jurado-Rodríguez et al., 2020a) manages the occlusion problem by estimating a triangle mesh from the k-nearest neighbors. Although it works on planar surfaces, canopy reconstruction is much more complex and constitutes a research field by itself.

We propose an automatic methodology that overcomes the described drawbacks of reconstructing thermal point clouds. The complete workflow is developed from scratch, enabling any further processing and optimization through hardware capabilities, e.g. using the Graphics Processing Unit (GPU). Firstly, a point cloud is reconstructed through SfM-MVS and high-resolution RGB imagery. Consequently, large and dense point clouds are used as the input of this approach. Then, coacquired RGB and TIR images are registered. The projection of a 3D point cloud into image space is finally performed considering a viable geometrical description of the scene. Furthermore, we evaluate the use of multiple aggregation functions to minimize the distance from aggregated values (3D point cloud) to image samples. As a result, we acquire a dense thermal point cloud that preserves image details. In addition, we solely need to manage a unique point cloud and thus GCPs are marked only once over RGB images, where they are undoubtedly visible.

Consequently, the main contribution of this work is the generation of large and dense thermal point clouds by mapping 3D points into 2D TIR images along with occlusion tests. Furthermore, we use the GPU hardware to solve time-consuming tasks in a reduced response time, outperforming currently available approaches for generating thermal point clouds. Hence, the presented method is a disruptive solution based on GPU computing for the generation of thermal 3D point clouds in order to help a better understanding of thermal distribution in real-world environments. The complete procedure of our mapping methodology is shown in Fig. 1.

This paper is structured as follows. The data acquisition method is first described as an introduction to our technologies. Second, we describe the details of our solution to construct 3D environments with RGB and thermal information. Then, we present the study area and assess our implementation through dispersion measures and frequency tests to prove the preservation of 2D color information in Section 3. The performance and results of our method are also evaluated with respect to commercial solutions. The outcomes of the conducted tests as well as the conclusions of this work are summarised in Section 4.

2. Materials and methods

2.1. Data acquisition

UAV imagery was acquired from a dual payload device stabilized by a gimbal (DJI Zenmuse XT2, Fig. 2). The RGB camera provides high-resolution images (CMOS sensor with 12 MP), whereas the thermal sensor captures images of lower resolution (Table 1). Regarding the sensor radiometric calibration, most metadata parameters are pre-calibrated by the manufacturer, while a few values can be configured through in-situ measurements to improve temperature estimations. Hence, the environmental temperature is measured and set as the background temperature to replace the default value. Flat Field Correction (FCC) is also performed before the flight to enhance image quality. Nevertheless, a detailed evaluation of the calibration accuracy is out of the scope of this work since our method is focused on the fusion of RGB and thermal data in 3D point clouds. The thermal sensor is configured to represent TIR results by means of a grayscale palette.

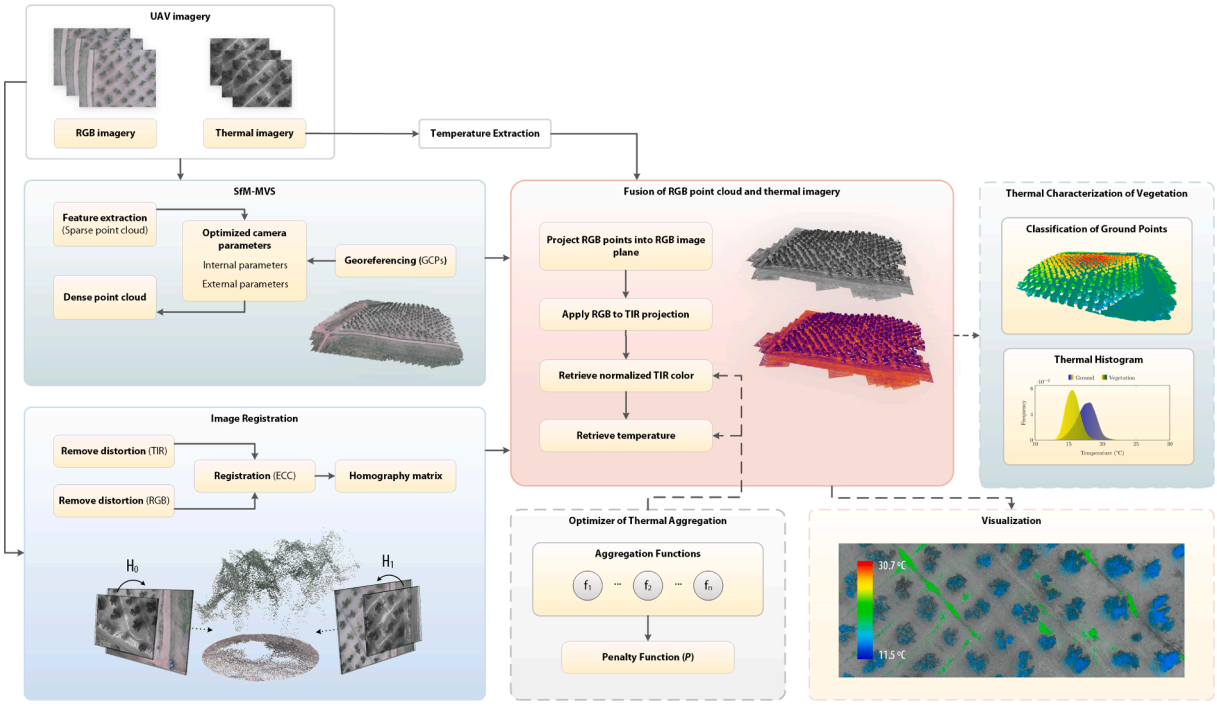

Fig. 1. Summary of the procedure of this work. SfM-MVS and image registration stages build isolated results, and therefore, can be performed in parallel, whereas fusion of 3D and 2D information depends on previous stages. Thermal data is assigned to 3D points by considering the point cloud occlusion and selecting an appropriate aggregation operator to reduce the error from image samples to the outcome. The visualization of the resulting point cloud is improved by highlighting outlier values. Furthermore, we describe a naive segmentation algorithm to characterize vegetation through its temperature.

Fig. 2. Unmanned aerial vehicle, DJI Matrice 210 RTK, and thermal camera, DJI Zenmuse XT2, in the area of study.

Table 1
Specifications of imaging sensors.

	Sensor	
Attributes	RGB	Thermal Infrared
Resolution	4000 × 3000 pixels	640 × 512 pixels
Focal Length	8 mm	19 mm
File Format	JPG	RJPG
FOV	57.12° × 42.44°	32° × 26°
GSD	0,6885 cm/pixel	4,57242 cm/pixel

Visible light and thermal sensors are mounted on a quadcopter drone (DJI Matrice 210 model). Despite both types of images are acquired synchronously, there exists a delay between both frames. Furthermore, drone movement and distance of sensors generate a more complex description of such misalignment. As a reference, Fig. 3 depicts the area covered by two co-acquired RGB and thermal images.

The UAV flight was performed in November 2019, whereas the mission was planned and executed using DroneDeploy (California, CA, USA) in a remote control device. The flight time was set to 19 min with a fixed altitude of 45 m (695 m above sea level) from take-off position. The planned UAV path defines a Boustrophedon path, i.e. back and forth

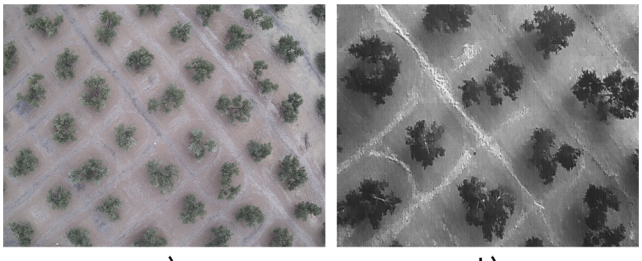

Fig. 3. Coacquired images from the dual device. a) RGB image, b) thermal image.

parallel lines. The view direction was configured as nadir, with a frontal overlap of 90% and a side overlap of 85%. Overlap values were increased to avoid issues as a consequence of elevation changes. As a result, a total of 820 images were acquired, i.e. 410 images for each sensor. Moreover, the network of GCPs was established prior to the UAV flight by distributing them on the edges as well as inside the study area (Martínez-Carricondo et al., 2018). In addition, GCPs are sparsely placed

on points with significant height variations, given the non-uniform plot elevation.

2.2. Point cloud estimation

RGB point clouds are acquired by combining high-resolution RGB images through methods that estimate three-dimensional structures from sequences of images. However, this procedure does not represent a challenge, as multiple libraries and software solutions provide implementations for methods such as SfM-MVS (Structure from Motion/Multi-View Stereo) (Webster et al., 2018). This technique is based on finding features that can be recognized on multiple images, so that they can be aligned. The recognition of key-points allows estimating external and internal camera parameters, enabling the calibration of the whole set of images. Key-points also represent an initial approximation of the point cloud through a sparse reconstruction. Both sparse point cloud and estimated camera parameters can be further optimized through bundle adjustment and the marking of GCPs in a real-world coordinate system, e.g. UTM.

A dense point cloud is finally calculated through MVS algorithm. The density and storage size of the result depends on the image resolution. Hence, this is an adjustable parameter that is later used by those methodologies dealing with the occlusion problem in the point cloud.

Table 2 shows the nomenclature of the parameters used throughout this work, some of them calculated through the feature recognition process of SfM-MVS. Others are previously known from the device specifications (e.g. sensor width). Even though the resolution is variable during SfM-MVS procedure, these values are calculated from the initial size.

RGB images are suitable for estimating 3D structures due to their high resolution and color space. However, reconstructing a point cloud from thermal images is not trivial as their resolution is low and they are frequently noisy (Sledz et al., 2018). The aforementioned problems harden the extraction of key-points due to the inconsistency of their color space. Consequently, point clouds calculated from these images may contain noticeable errors, e.g. empty areas that could not be reconstructed. In addition, the stage of bundle adjustment of SfM-MVS can not be optimized accurately through the marking of Ground Control Points (GCP) due to an inadequate color consistency for human operators. As a result, a naive solution based on aligning both point clouds through an algorithm such as ICP can be hard to manage using erroneous TIR point clouds with low point density.

Table 2
Parameters obtained both from RGB and thermal camera calibration, as a pre-processing stage for SfM algorithm, and from image metadata.

Parameter	Definition
Image size (w_{image}, h_{image})	Original size of images.
Principal point (c_x, c_y)	Intersection of principal axis and image plane.
Focal length (f_x, f_y)	Distance in pixels from the center of projection and the image plane.
Width and height of sensor (w_{sensor}, h_{sensor})	Size in millimeters of both dimensions.
Omega, Phi, Kappa (ω, ϕ, κ)	Rotation between image coordinate system and world system.
Camera position (t_{local})	Camera position in point cloud system.
World offset (t_{world})	Offset between local coordinate system and UTM system.
Camera matrix (K)	Calculated from previous parameters: $\begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}$
Rotation matrix (R)	Composition of $R(\omega)\cdot R(\phi)\cdot R(\kappa)$.
Projection matrix (P)	Distortion-free projection of a pinhole model: $K\cdot[R\mid -Rt_{local}]$.
Radial distortion (k_1, k_2, k_3)	Distortion mostly visible on straight lines.
Tangential distortion (p_1, p_2)	Misalignment of device lens with respect to the image plane.

2.3. Thermal image processing

This section operates with thermal images to correct them and extract derived information. Results can be calculated once and stored to reduce response time for the following executions of our software solution.

2.3.1. Geometric calibration

The projection of estimated points on the image plane works under ideal images. However, radial distortion is observed when using lenses with small field of view. Such aberration can be corrected through the camera matrix, K, and both radial and tangential distortion coefficients, $(k_1, k_2, p_1, p_2, k_3)$. Such factors are not provided as image metadata, and therefore are determined as part of the bundle adjustment of SfM by considering the image data.

The dominant value for radial distortion coefficients is k_1. This value is smaller than zero for thermal images, i.e. they present a pincushion distortion where lines bow inwards. Consequently, correcting such distortion generates images with blank values as long as the original dimensions are preserved. Therefore the minimum area with non-null color information must be calculated to crop every image. As a result, corrected thermal images end up with lower resolution than expected. Fig. 4 compares three stages of the correction procedure.

Corrected images are calculated through Eq. 1 (Mallon and Whelan, 2004) by solving the problem of inverse distortion correction, where undistorted images are created from a distorted image (de Villiers et al., 2008). Consequently, each undistorted pixel is guaranteed to be mapped with a distorted pixel.

$$[\bar{x}, \bar{y}, \bar{z}]^{\intercal} = \left[\frac{x - c_x}{f_x}, \frac{y - c_y}{f_y}, 1\right]^{\intercal} = \boldsymbol{K}[x, y, 1]^{\intercal} \tag{1}$$

$$r^2 = \bar{x}^2 + \bar{y}^2 \tag{2}$$

$$x_u = \bar{x}\cdot\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right) + 2p_1\bar{x}\cdot\bar{y} + p_2\left(r^2 + 2\bar{x}^2\right) \tag{3}$$

$$y_u = \bar{y}\cdot\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right) + p_1\left(r^2 + 2\bar{y}^2\right) + 2p_2\bar{x}\cdot\bar{y} \tag{4}$$

Note that the pair (x_u, y_u) describes a non-distorted point in the corrected image, while ideal images are generated in the opposite manner. We need to determine for each pixel, defined by integer coordinates, the distorted pixel to use. Eq. 5 defines the inverse procedure to determine $x_d, y_d \in \mathbb{R}$. Consequently, an interpolation function such as the bilinear function is applied in this procedure.

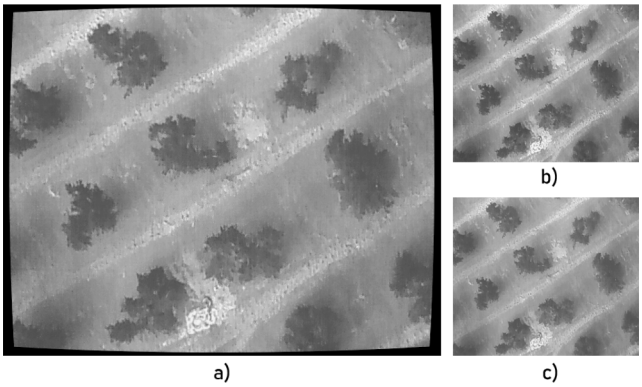

Fig. 4. Comparison of a) free of distortion thermal image with blank values preserving the original size, b) distorted thermal image and c) free of distortion thermal image with reduced size.

$$[x_d, y_d, 1]^\intercal = [\overline{x}f_x + c_x, \overline{y}f_y + c_y, 1]^\intercal = \\ = K[\overline{x}, \overline{y}, 1]^\intercal \quad (5)$$

2.3.2. Temperature computation

Thermal images show a normalized representation of temperature captured on the scene. Though, absolute values are not known from color data. However, automatic computation of temperature can be performed through radiometric file formats, e.g. RJPG. Thermal cameras record the radiation emitted from an object surface, either it comes from the object itself or surrounding objects. However, it is affected by several environmental parameters, such as the air humidity, the air temperature and the background temperature (beyond the object reflectivity and emissivity, as well as the distance between the surface and the camera) (DJI, 2018). Therefore, each manufacturer models the transmissivity of the atmosphere through a theoretical or empirical law, using some constant values embedded in the image metadata (Teza and Pesci, 2019). Consequently, we need to extract several parameters from the embedded data, including the emissivity, atmospheric temperature, reflected apparent temperature, infrared window temperature, infrared window transmission, relative humidity, Planck's constants (P_{R1},P_{R2},P_O, P_F, P_B) and atmospheric transmission constants (α_0, α_1, β_0, β_1, X). Nevertheless, most of them are calibrated by the manufacturer. Hence, temperature T is given by Eq. 6:

$$T = \frac{Planck_B}{ln\left(\frac{Planck_{R1}}{Planck_{R2}(radiance+Planck_O)} + Planck_F\right)} \quad (6)$$

where *radiance* is given by Planck's Law for every pixel. We refer the reader to (Minkina and Dudzik, 2009; Teza and Pesci, 2019) for further details.

Absolute temperature values for every image are calculated once and stored to be processed by our solution. Fig. 5 shows the thermal boundaries of an image. Temperature values have also been normalized and presented through a mapping function defined by a linear interpolation and the texture on the right of Fig. 5.

2.4. RGB image processing

Previous distortion concepts can also be applied in this step. RGB images present geometric distortions since they are captured by a wide-angle lens. However, their distortion model produces the barrel effect instead of the pincushion defect. In consequence, the dominant radial distortion k_1 is greater than zero, although the image can be corrected by using Eqs. 1 and 5 as shown previously. Both tangential and radial coefficients are extracted from the first stage of SfM-MVS as well.

Fig. 6 compares an RGB image with barrel distortion, where lines are bowed outwards, and the corrected result. As opposed to the pincushion effect, reversing this distortion produces images of the same resolution, although the captured area is not completely preserved.

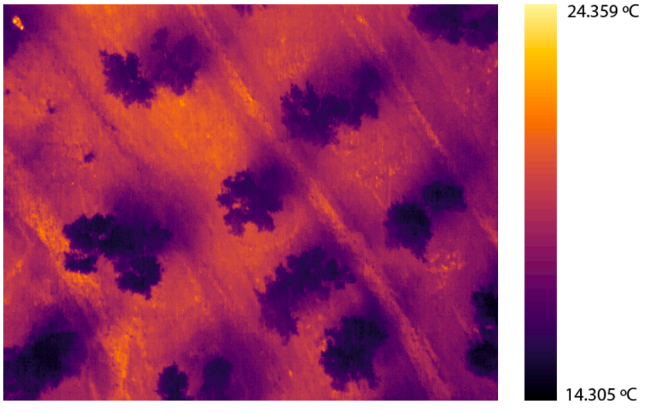

Fig. 5. TIR image coloured with a mapping function. Absolute values can be observed through the represented maximum and minimum values.

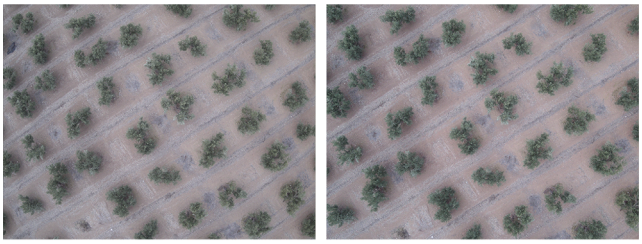

Fig. 6. Comparison of a) distorted RGB image and b) RGB image without distortion.

2.5. RGB and thermal image registration

The fusion of RGB point clouds with thermal data can be performed through image registration. This solution avoids errors derived from the estimation of thermal point clouds. Furthermore, our solution manages a single 3D point cloud where processing tasks are carried out (e.g. georeferencing through multiple GCPs). For this purpose, 3D RGB points need to be projected into the RGB image plane, where this fusion occurs. This transformation is defined as follows (Eq. 7, 8):

$$[x', y', z']^\intercal = P[x, y, z]^\intercal \quad (7)$$

$$[x'', y'']^\intercal = \frac{1}{z'}[x', y']^\intercal \quad (8)$$

provided that $[x, y, z]$ is expressed in the local coordinate system where both the point cloud and the cameras are defined. Otherwise, if defined in a global coordinate system such as UTM:

$$[x, y, z]^\intercal = [x_{global}, y_{global}, z_{global}]^\intercal - t_{world} \quad (9)$$

Consequently, each 3D point is associated with two values, $x'', y'' \in \mathbb{R}$, for each image of the dataset. Note that t_{world} is defined as an arbitrary shift of real-world points to avoid working with large values. The same reasoning applies to local positions of cameras (t_{local}).

On the other hand, the registration of RGB and thermal imagery has been previously achieved with the Enhanced Correlation Coefficient (ECC) algorithm (López et al., 2021; Evangelidis and Psarakis, 2008) so that the misalignment of two images is expressed through a transformation matrix, H_i, of variable complexity. Hence, H_i is the result of an optimization problem whose objective function is provided by a correlation coefficient for both template and input image. In our solution, H_i is defined as a homography matrix of size 3x3, which is proved enough to describe perspective differences between images.

The procedure for applying ECC to RGB and TIR images starts by cropping each RGB image. However, the homography matrix H_i also includes a rotation for some pairs of images due to capture delays and platform movement. Consequently, the area to be cropped is expressed as an approximate rectangle centered at $\left(\frac{w_{image}}{2}, \frac{h_{image}}{2}\right)$ that guarantees to cover TIR images under most of the valid rotations. Therefore, RGB images are defined as source images, while TIR images are transformed to align both of them. The computed homography matrix allows projecting thermal images into cropped RGB images, whereas the inverse matrix enables the backward projection.

In summary, Eqs. 7 and 8 allow to project 3D RGB points to RGB images, while $C_i \cdot H_i$ projects 2D RGB points into a TIR image plane, provided that C_i is a composite matrix (Eq. 10) that defines the inverse cropping transformation, so that RGB pixels are projected into thermal images. However, note that projected points may not be within the

rectangle defined by the size of undistorted TIR images. Nevertheless, the inverse homography matrix H_i leads to an ideal rectangle, and thus the 'point in polygon' test is carried out by four Boolean operations instead of more complex solutions (Feito et al., 1995).

$$
\begin{aligned}
C_i &= T_{crop} * S_{ratio} * T_{-center_{RGB}} = \\
&= \begin{bmatrix} 1 & 0 & \frac{w_{thermal_u}}{2} \\ 0 & 1 & \frac{h_{thermal_u}}{2} \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} r_x & 0 & 0 \\ 0 & r_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \\
&\cdot \begin{bmatrix} 1 & 0 & -\frac{w_{RGB_u}}{2} \\ 0 & 1 & -\frac{h_{RGB_u}}{2} \\ 0 & 0 & 1 \end{bmatrix} = \\
&= \begin{bmatrix} r_x & 0 & \frac{1}{2}\left(- r_x w_{RGB_u} + w_{thermal_u} \right) \\ 0 & r_y & \frac{1}{2}\left(- r_y h_{RGB_u} + h_{thermal_u} \right) \\ 0 & 0 & 1 \end{bmatrix} \\
&\left(r_x, r_y \right) = \left(\frac{crop_x}{w_{thermal_u}}, \frac{crop_y}{h_{thermal_u}} \right)
\end{aligned}
\tag{10}
$$

where ($crop_x$, $crop_y$) are the dimensions of the cropped area, ($w_{thermal}$, $h_{thermal}$) are the dimensions of undistorted thermal images and u subscript refers to the index of an undistorted image. Fig. 7 shows the result of the registration methodology, where the warped quadrilateral shape is also represented.

Composite matrices C_i are calculated once and stored in binary files since this process is time-consuming. Due to the complexity of the alignment methodology, there may exist pairs of images that can not be registered as intensity highly differs between both types of images. In such a case, the ECC algorithm fails to converge or the resulting quadrilateral shape highly differs from an ideal rectangle. Both erroneous scenarios can be easily observed, although the second detection relies on a strict threshold based on minimum/maximum angles degrees enclosed by the rectangle shape. However, those 3D points visible on discarded images more than likely appear in multiple images as the drone flight mission is planned so that most areas are covered by five images at least. Table 3 shows the percentage of thermal images which can be matched to RGB images for our dataset as well as the number of 3D points that were not visible from any registered thermal image. Note that the field of view is wider for our RGB device, therefore edge points are also discarded as they are not visible from any thermal image. Consequently, it is observed that all the unregistered points are also labeled as points not visible from any viewpoint (edge points). For larger point clouds, the number of edge points grows unevenly with respect to the rest of the scenario. Hence, the percentage of unregistered and edge points decreases.

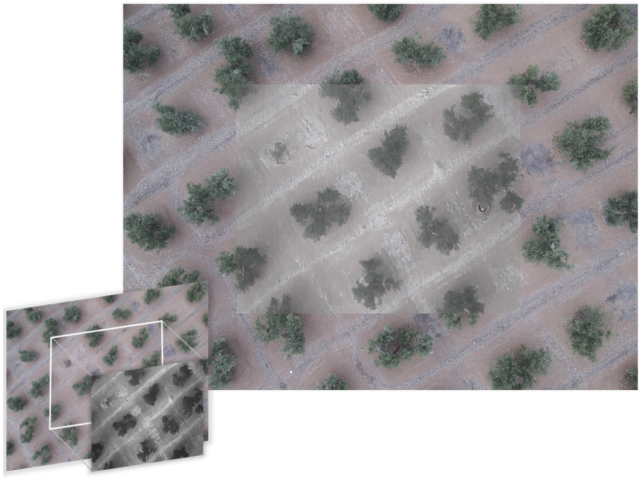

Fig. 7. A result from the registration of RGB and thermal images. The thermal image is overlapped within the RGB image and displayed with transparency ($\alpha = 0.7$). The bottom image presents the final quadrilateral shape.

2.6. Thermal image mapping

Merging RGB point clouds and TIR data involves the previously described projections. First, we implement a naive approach without any consideration about occlusion or visibility. As a result, 3D points are considered to be visible from a viewpoint as long as they are mapped to 2D points with valid thermal data.

Each thermal image presents two layers, both for relative temperature (original format) and absolute values (extracted in previous sections). Eqs. (7)–(9) allow projecting each 3D point into the RGB image plane. Points are considered not visible from a viewpoint and thus discarded whether their projected coordinates present values smaller than zero or larger than the dimensions of RGB images. In addition, the TIR registration within RGB images imposes another discard condition due to its smaller observed area.

From 2D coordinates ($x_{thermal}, y_{thermal} \in \mathbb{R}$) within TIR images, we can sample values through a bilinear interpolation in which surrounding pixels to $x_{thermal}, y_{thermal}$ also affect the outcome through weight factors given by their distance. The resulting values also contribute to an aggregation function that combines information from different TIR images for a 3D point, e.g. the arithmetic mean.

The main challenge during this step is the search of candidate points for each image. Regular grids can manage such search whether the set of points is distributed uniformly along the scenario, which is not assumed in our solution. Hence, an adaptive data structure is implemented to speed up the search. The point cloud is managed with an octree optimized for searching neighbors within a radius (Behley et al., 2015). A radius search is selected over a box search since the previously described methods to correct geometric distortions also apply interpolation functions. Therefore, the color of image edge points is considered to be less reliable. As a consequence, this radius represents a new condition that may also discard edge points, although this restriction can be avoided using a large radius. This data structure organizes the 3D RGB points instead of the array of image viewpoints, otherwise, an efficient search would not be needed due to the relatively low number of images. Note that the search radius is more intuitive when defined at ground level, and therefore it is always expressed relative to the ground point (x,

Table 3
a) Percentage of pairs of RGB-thermal images that were successfully registered. b) Percentage of points which were not visible from any thermal image for two point clouds with different size. Edge points represent those points that are not visible even when the complete image dataset is successfully registered. A radius r of 80 m was utilized during the search of candidate points to avoid omitting visible values.

a) Two-dimensional space

Number of pairs	Resolution	Registration success
410	640 × 512 pixels	89,7561%

b) Three-dimensional space

Number of points	Unregistered	Edge points
19.161.076	18,8952%	18,8952%
98.016.324	15,5823%	15,5823%

$aabb_{center_y}, z)$, where x and z depends on the local position of the image viewpoint and $aabb_{center_y}$ is the y coordinate for the center of the axis-aligned bounding box (AABB) of the environment. Fig. 8 renders this data structure as a set of spheres (optimized octree) and as a set of bounding boxes (the conventional representation of an octree).

Fig. 9 shows a top view of the reconstructed thermal point cloud. Close-up images emphasize the preservation of details which are visible on thermal images, such as rocks or paths.

2.7. Occlusion problem

The previous approach presents a comprehensible methodology for reconstructing a thermal point cloud, although it is not geometrically accurate. The color of occluded points can not be estimated by considering the data of foreground objects. Previous studies have used ray-casting along with voxelization to solve this problem (Vidas et al., 2015). However, this solution lacks accuracy and flexibility as octrees are mainly constructed with top-down methodologies, i.e. the size of a voxel is determined by the subsequent subdivisions of the point cloud bounding box (AABB) and the octree level.

Here we propose multiple solutions to produce a more color-realistic point cloud. Then, the outcomes are compared to visualize their fidelity with respect to our image data-set.

2.7.1. Visibility test

Images represent a discrete domain from where the SfM-MVS algorithm can estimate at least one 3D point from an RGB pixel. For each image we can build a depth buffer (or z-buffer) of the same size as the resolution used during the SfM estimation. Nevertheless, we can increase the size of the depth buffer whether we consider sub-pixel approaches of SfM. This procedure saves the closest point to the current viewpoint if multiple 3D points are mapped to a pixel. Therefore, the depth (z) determines if a point is visible. Depth buffers can be represented as a sparse matrix instead of a matrix of fixed size to reduce memory usage, mainly for high resolutions. This sparse matrix is implemented as a hash map that stores pairs of values; their key is given by the access index of a 2D point and the value is the nearest point for such position. The depth buffer can be rendered as shown in Fig. 10, where a circular pattern is visible as a result of the search algorithm. The workflow for determining visible points is presented in Algorithm 1.

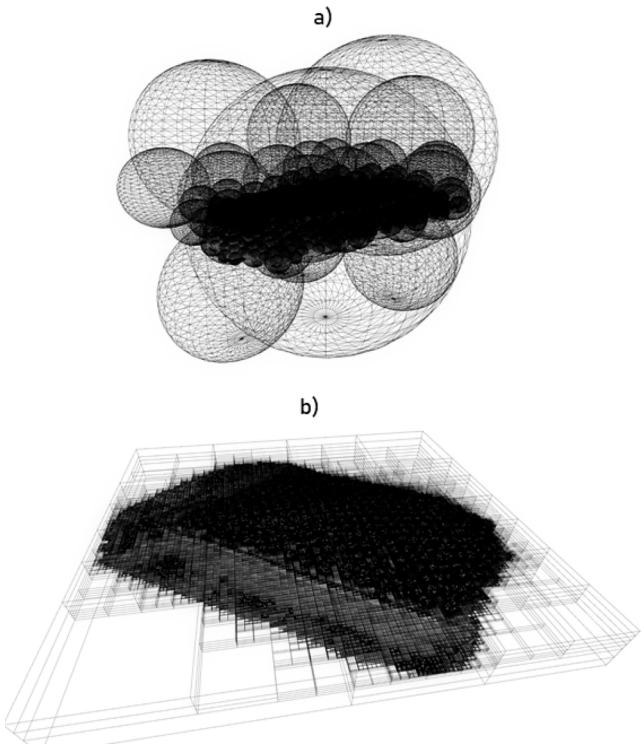

Fig. 8. Rendering of a) an octree whose nodes are spheres, optimized for searches within a radius and b) a basic octree whose nodes are boxes.

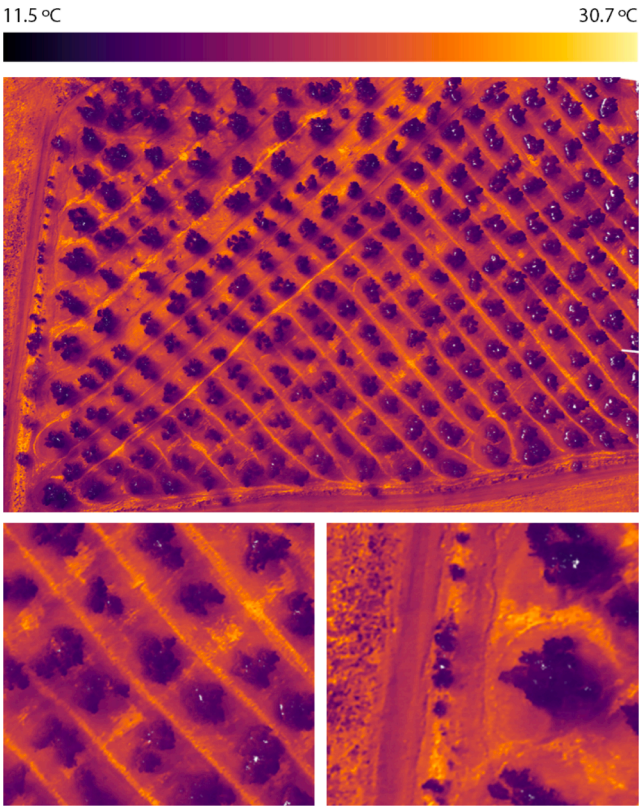

Fig. 9. Top view of a thermal point cloud captured by a nadir camera. Absolute temperature values are normalized and mapped to the upper texture. The search radius was fixed to 30 m.

Algorithm 1. Visibility test to accurately map thermal images into an RGB point cloud.

Input RGB point cloud: P
Input RGB images: c
Input RGB resolution during SfM-MVS procedure: (w_{SfM}, h_{SfM})
Input Search radius: R
Output Array of normalized and absolute thermal values
1: **for** Every image c_i captured from the position p_{c_i} **do**
2: Create a new buffer of size (w_{SfM}, h_{SfM}) with $d_{x,y} = \infty$ provided that $0 \leqslant x < w_{SfM}$ and $0 \leqslant y < h_{SfM}$.
3: Retrieve candidate points within the radius R through the point cloud octree
4: **for** Every three-dimensional candidate point p_j **do**
5: Calculate 2D TIR point (x, y) (Eqs. 7, 8)
6: **if** (x, y) in TIR polygon **then**

(*continued on next page*)

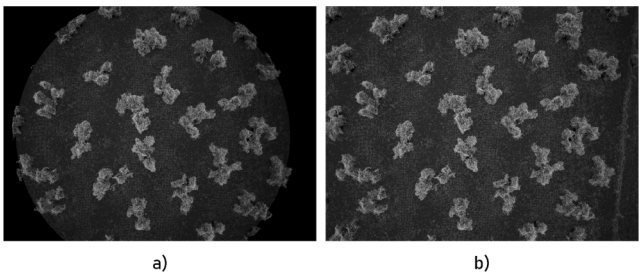

Fig. 10. Depth buffer of a camera during a visibility test carried out with a radius of size a) 20 m, b) 30 m.

(*continued*)

```
        if distance(p_ci, p_j) < d_x,y then
          Add point to depth buffer and update the minimum distance found at d_x,y
        end if
      end if
    end for
    for Every point p_j stored in depth buffer do
      Retrieve and accumulate the normalized and absolute thermal values from c_i for p_j
    end for
end for
Compute final values for each point of P as an arithmetic mean
```

2.7.2. Occlusion test

Occlusion tests can be ideally managed through a ray-casting implementation as long as the target structure is presented as a triangle mesh (Jurado-Rodríguez et al., 2020a). However, estimating the mesh from a point cloud is a challenging task. Multiple algorithms are found in the bibliography; some require the normal for each point, while others are robust enough to work without surface orientation, e.g. Advancing Front reconstruction (Cohen-Steiner and Da, 2004). Furthermore, surface reconstructions are commonly based on time-consuming algorithms that rely on multiple parameters non-consistent along different point clouds, e.g. a search radius. In addition, the outcomes are not accurate for the cumbersome problem of reconstructing vegetation models (Zhang et al., 2014; Gong et al., 2018). Fig. 11 shows the result of Advancing Front surface reconstruction for a subset of RGB points. Faces are oriented considering that the viewpoint is above the scene. Hence, faces are flipped if $\hat{n}\cdot(\widehat{v-p}) > \frac{\pi}{2}$, provided that $\hat{n}$ is the surface normal, v is an image viewpoint and p is a 3D point over the triangle mesh surface. A schematic representation of our occlusion methodology is depicted in Fig. 12.

However, points can be occluded by others if they have a volume. This assumption allows solving this problem through ray-casting, though solving the occlusion problem for millions of points is a time-consuming task. Some data structures can discard large parts of the scene with each advancing step, such as a Bounding Volume Hierarchy (BVH), a tree structure where each primitive of the scene is bounded by an AABB, while surrounding AABBs are merged up to the tree root. Regarding the tree quality, better cluster separations lower the trace time as they allow discarding larger parts of the scene, thus reducing the number of steps needed to traverse the scene.

In this work, point volumes are modeled as spheres to provide an efficient alternative to viewpoint oriented boxes. Otherwise, the quality of the BVH worsens due to self-intersecting geometry (AABBs from a BVH are axis-aligned instead of object-oriented), whereas the intersection test of an oriented bounding-box (OBB) with a ray is slightly more

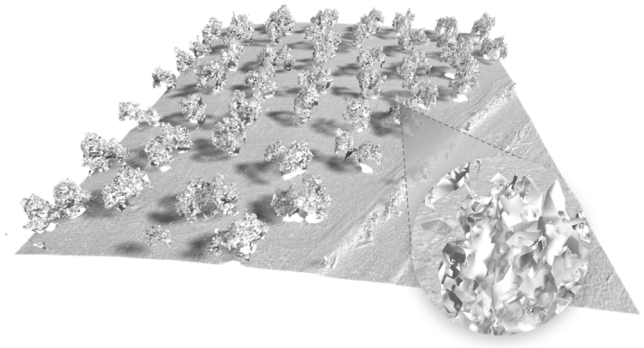

Fig. 11. Triangle mesh estimated for a subset of our point cloud through Advancing Front algorithm. Reconstruction of vegetation is clearly erroneous in comparison with the ground.

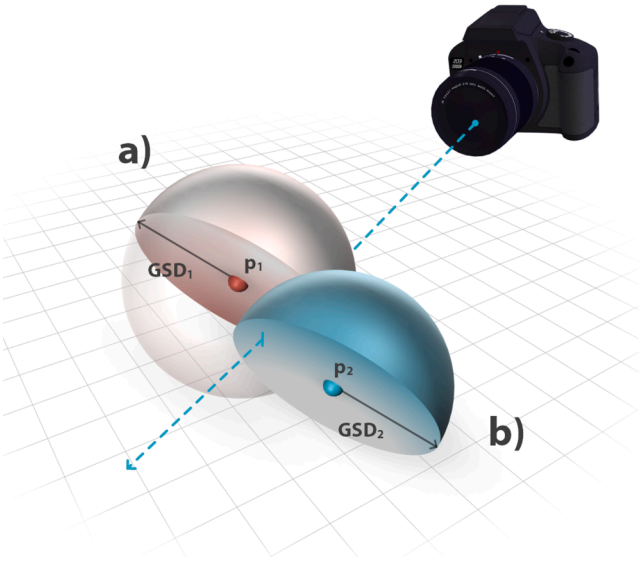

Fig. 12. Representation of an occlusion test for two points expressed as spherical volumes. a) Point where the ray impacts first (p_1), occluding the point p_2, b).

complex. The sphere radius depends on the relative height from a point to an image viewpoint (Fig. 12), and it is calculated as defined in Eq. 11.

$$R = \frac{\frac{w_{sensor}\cdot height}{f_x\cdot w_{image}}}{2} = \frac{\frac{h_{sensor}\cdot height}{f_y\cdot h_{image}}}{2} = \frac{GSD}{2} \tag{11}$$

with *GSD* being the Ground Sample Distance, i.e., the size of a pixel when target is at an altitude of *height*. *R* is defined in meters per pixel, although w_{sensor} and f_x are defined in millimeters.

Note that spherical volumes vary for each 3D point and image viewpoint, and therefore BVH trees need to be built for every subset of candidate points. However, we implement their construction on GPU hardware to speed up the procedure through the methodology defined in (Meister and Bittner, 2018). This work proposes a spatial ordering of primitives using their Morton codes (30 bits that represent *xyz* coordinates) and merging nodes by searching efficiently the nearest neighbour (which also happens to be near within the ordered buffer). On the other hand, the ray-casting problem is also solved on GPU using the BVH data structure. For that purpose, we generate a ray for each 3D point. Occluded points receive a collision linked to a different volume, whereas visible points find themselves as the nearest volume to the viewpoint (Fig. 12). A schematic workflow for this methodology is presented in Algorithm 2.

This algorithm generates point clouds where their thermal values are less likely to be calculated from as many images as the naive approach. Furthermore, we avoid contributions from images where points are not visible, although it also tends to discard more points as some pairs of images are not registered successfully.

Algorithm 2. Occlusion test to determine temperature values of sphere-shaped RGB points with adaptive radius.

```
Input RGB point cloud: P
Input RGB images: c
Input Search radius: R
Output Array of normalized and absolute thermal values
for Every image c_i captured from the position p_ci do
  Retrieve candidate points within the radius R through the point cloud octree
  Build a new BVH with the array of candidate points
    Sort candidate points expressed as Morton codes and build BVH leaves
    Join leaf and intermediate nodes while root is not reached
  for Every three-dimensional candidate point p_j do
    Throw a ray towards p_j and find the first intersected volume
```

(*continued on next page*)

(*continued*)

```
    if Intersected volume = volume_pj then
      Calculate 2D TIR point (x,y) (Eqs. 7, 8)
      if (x,y) in TIR polygon then
        Retrieve and accumulate the normalized and absolute thermal values
  from c_i for p_j
      end if
    end if
  end for
end for
Compute final values for each point of P as an arithmetic mean
```

2.8. Visualization

This section describes a method for improving the visualization of thermal data. (Javadnejad et al., 2020) proposes a linear interpolation between RGB and TIR data, where RGB colors are converted to grayscale and temperature values are mapped to an RGB value through a color-ramp function. The weight, $w(t)$, is computed as the distance of each temperature sample, t, to the mean, t_{mean}, so that $w(t) = 1$ omits RGB colors. Although it accentuates cold and hot regions ($w(t) \leftarrow 1$), intermediate values are also merged with grayscale colors. Therefore, the representation of non-anomalous temperature values distorts the color distribution of the RGB point cloud and does not help to identify anomalies.

We detect outlier values through the conventional numeric outlier method. For that purpose, thermal values were sorted in GPU to retrieve first and third quartiles, Q_1 and $Q3$, as well as the interquartile range ($IQR \leftarrow Q3 - Q1$). Values under $Q_1 - k(IQR)$ and above $Q3 + k(IQR)$, where $k \geqslant 0$, are considered to be anomalous temperature values and thus they take $w(t) \leftarrow 1$. The visualization of intermediate values is computed as proposed in (Javadnejad et al., 2020). However, RGB colors are not converted to grayscale nor we utilize a linear interpolation. Instead, we use a Hermite interpolation that emphasizes values of t close to t_{min} and t_{max}.

Eq. 12 shows the procedure for computing the proposed enhanced visualization. Note that γ allows us to modify the brightness of RGB colors, where $0 \leqslant \gamma \leqslant 1$. Fig. 13 depicts the use of γ to darken irrelevant regions. On the other hand, $\zeta(t)$ represents a color mapping function that is also parameterized by the temperature. In this work, $\zeta(t)$ is given by the texture depicted at the left part of Fig. 13.

$$\begin{bmatrix} r' \\ g' \\ b' \end{bmatrix} = H\left(w\left(t\right)\right)\zeta\left(t\right) + \gamma\left(1 - H\left(w\left(t\right)\right)\right)\begin{bmatrix} r \\ g \\ b \end{bmatrix}$$

$$w\left(t\right) = \begin{cases} 1 & t_{min} \leqslant t < Q_1 - k(IQR) \\ \dfrac{t_{mean} - t}{t_{mean} - t_{min}} & Q_1 - k(IQR) \leqslant t \leqslant t_{mean} \\ \dfrac{t - t_{mean}}{t_{max} - t_{mean}} & t_{mean} < t \leqslant Q_3 + k(IQR) \\ 1 & Q_3 + k(IQR) < t < t_{max} \end{cases} \tag{12}$$

$$H\left(w\left(t\right)\right) = \begin{cases} w(t) & w(t) = 1 \\ w(t)^2\left(3 - 2w\left(t\right)\right) & \text{otherwise} \end{cases}$$

2.9. Aggregation of image samples

3D points are mapped to several images; hence, we have many thermal measurements for each point. However, the acquired TIR radiation changes in relation to the viewing angle, expressed through azimuth, ϕ, and zenith angle, δ, given that $\phi \in [0, 360]$ and $\delta \in [0, 180]$. Any real surface emits radiation non uniformly in its hemisphere unless it is considered to be a Lambertian emitter (Vollmer and Möllmann, 2017). Furthermore, objects emit more radiation when $\delta = 0$, i.e., the viewing direction is normal to its surface. Fig. 14 depicts 19 TIR radiation samples acquired for the same 3D point with respect to the Lambertian behaviour, both of them scaled by a factor of π.

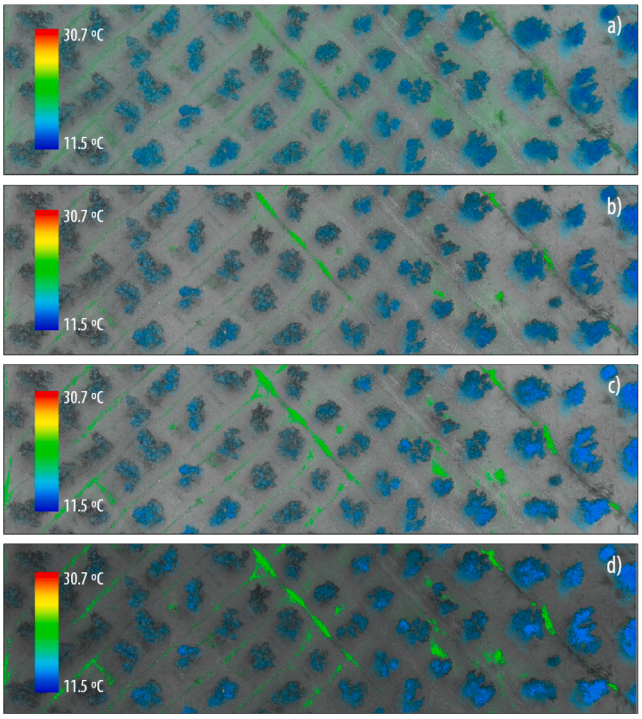

Fig. 13. Visualization of the study area using multiple configurations. a) Visualization proposed by (Javadnejad et al., 2020), while b), c) and d) show our modified procedure. b) $\gamma = 0.8, k = 0.8$, c) $\gamma = 0.8, k = 0.5$, d) $\gamma = .0.5, k = 0.5$.

Although the arithmetic mean has been previously assumed as the standard aggregation for any number of samples (Javadnejad et al., 2020), here we propose a procedure with multiple aggregation functions that can be either used separately or combined. When combined, their outputs are compared to a set of inputs to select the aggregation that minimizes the outcome of a penalty function (Bustince et al., 2017b; Bustince et al., 2017a). Consequently, the penalty function determines the similarity of each aggregation output to a set of TIR radiation samples.

The fusion of information by means of penalty functions has been

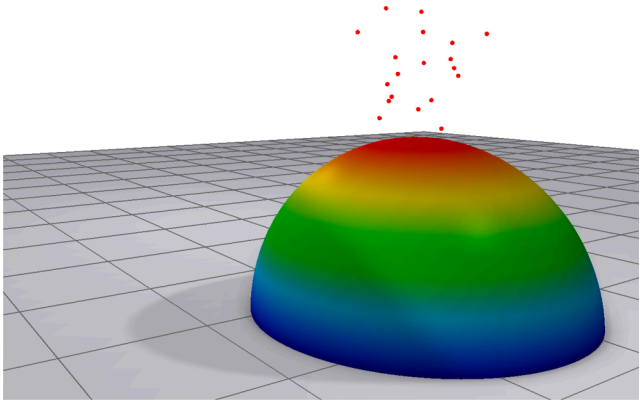

Fig. 14. Representation of a point cloud with 19 TIR radiation samples observed from different viewpoints for the same surface point. Sampled emission values are compared to a Lambertian radiator (semisphere).

gaining interest in scientific research. However, it has hardly been explored in the field of computer graphics (Paternain et al., 2012). Furthermore, the correct implementation of this procedure allows to omit penalty functions when needed and exchange the arithmetic mean for another suitable operator. Note that TIR radiation samples are acquired from different points of view; hence, these values are consistent by themselves. Therefore, a function that selects one value is also valid, e.g. *maximum* or *minimum* operators as well as any behaviour that establishes a preference for a viewing angle.

Briefly, a function is called an n-ary aggregation function $f : [0,1]^n \rightarrow [0,1]$ if it is monotone increasing and satisfies the boundary conditions (Eq. 13).

$$\begin{gathered} \forall i \in \{1,\ldots,n\},\ \text{if } x_i \leqslant y \Rightarrow \\ \Rightarrow f(x_1,\ldots,x_n) \leqslant f(x_1,\ldots,x_{i-1},y,x_{i+1},\ldots,x_n) \\ f(x_{min},\ldots,x_{min}) = x_{min} \\ f(x_{max},\ldots,x_{max}) = x_{max} \end{gathered} \tag{13}$$

Furthermore, it is an averaging aggregation function if it is bounded by the minimum and maximum input (Eq. 14).

$$min\{x_1,\ldots,x_n\} \leqslant f(x_1,\ldots,x_n) \leqslant max\{x_1,\ldots,x_n\} \tag{14}$$

Hence, the local penalty function is expressed as the function $P : \mathbb{R}^2 \rightarrow \mathbb{R}$ that satisfies the conditions defined in Eq. 15, given that y is the aggregated value. Consequently, the penalty based function is summarized as $\sum_{i=1}^{n} P(x_i,y)$, where the objective is to retrieve the aggregation that minimizes the value of $P, g(x) \leftarrow \text{argmin}_y P(x,y)$.

$$\begin{gathered} P(x_i,y) = 0 \quad \forall x_i = y \\ P(x_i,y) > 0 \quad \forall x_i \neq y \\ P(x_i,y) \geqslant P(x_j,y) \ \text{ if } \ |x_i - y| > |x_j - y| \end{gathered} \tag{15}$$

However, our environment simplifies these expressions as $n = 1$ neither we require the use of (r,g,b) tuples. Normalized and absolute TIR radiation are represented with one value. Therefore, we simply need to define a set of aggregation and penalty functions, like the operators which are following enumerated.

i) Minimum: $f(x) = \text{argmin}_i x_i$
ii) Maximum: $f(x) = \text{argmin}_i x_i$
iii) Arithmetic mean: $f(x) = \frac{\sum_{i=1}^{n} x_i}{n}$
iv) Geometric mean: $f(x) = \sqrt[n]{\prod_{i=1}^{n} x_i}$
v) Harmonic mean: $f(x) = n\left(\sum_{i=1}^{n} \frac{1}{x_i}\right)^{-1}$

whereas penalty functions are less intricate as their objective is to measure the error between the inputs and the aggregated value, y. The following list presents three penalty functions (Paternain et al., 2012) here considered for the minimization problem. Note that we have omitted to include weighted aggregation operators, such as OWA (Ordered Weighted Averaging) (Zhou et al., 2010), or parameterized aggregations, for example, Bonferroni mean (Yager, 2009), as they imply more computational complexity. OWA operator requires sorting the samples, while parametric operators require adjusting variables to minimize the penalty function.

i) $P_1(x_i,y) = |x_i - y|$
ii) $P_2(x_i,y) = (x_i - y)^2$
iii) $P_3(x_i,y) = |x_i - y|^3$

Accordingly, the dissimilarity of new results and input samples is clearly reduced by minimizing an error function. We have measured the frequency of choosing each aggregation for different penalty functions. Table 4 shows the selection ratio of three proposed local penalty functions, $P(x_i,y)$, against five aggregation operators. As shown in Fig. 14, thermal values are not evenly distributed along a given range, and therefore the arithmetic mean is not always the preferred solution. However, maximum and minimum operators are seldom utilized; their highest frequency is observed for those procedures that discard occluded points and thus aggregate fewer values.

From the results, we can observe that the arithmetic mean is the most used aggregation, although the geometric and harmonic mean are also widely utilized. *maximum* and *minimum* are rarely selected, but their use is significantly increased for occlusion-based methods, where the number of samples is reduced. However, for optimization purposes, we could reduce the number of operators by discarding uncommon aggregations.

2.10. Thermal characterization of vegetation

Analyzing the temperature of object surfaces often implies segmenting the environment, so that target structures can be further studied. Our datasets characterize a natural environment whose main components are soil and trees and thus we only need to isolate and discard ground points. Regarding the classification procedure, we propose a geometric approach based on the surface orientation of a point cloud.

Consequently, normal vectors are first computed. The normal estimation is performed with different algorithms, so we can later compare their results and execution time. The first algorithm is implemented on GPU through a KNN (k-nearest neighbors) search that finds those surrounding points that contribute to the normal estimation. KNN search is optimized with an approach previously used while constructing a BVH (Meister and Bittner, 2018). Points are spatially ordered along a z-curve; first, they are expressed as Morton codes, i.e. 30 bits codes (3D), and then they are sorted with the Radix Sort algorithm. This solution has been previously explored on high-performance works (Connor and Kumar, 2010; Jakob and Guthe, 2021) due to its multiple benefits. First, points to be accessed for each index are reduced to $2r$, where r is the search radius. Access time to memory is also reduced through spatial locality as 3D multidimensional arrays are transformed into sorted 1D arrays. Furthermore, the complete search methodology is highly

Table 4
Comparison of frequency of choosing an aggregation operator using three different penalty functions for each proposed mapping method. Frequency is normalized in $[0,1]$, therefore the sum of a row is equal to 1. The first and second choice of each configuration are highlighted in bold.

Penalty Function	Mapping algorithm	Aggregation functions				
		Arithmetic mean	Geometric mean	Harmonic mean	Maximum	Minimum
$g_1 : \text{argmin}_y \sum_{i=1}^{n} \|x_i - y\|$	Naive Mapping	**0,64699**	0,07563	**0,23348**	0,02530	0,01858
	Depth Buffer Mapping	**0,65672**	0,06017	**0,21589**	0,03815	0,02905
	Occlusion Mapping	**0,67193**	0,03362	**0,17048**	0,07004	0,05392
$g_2 : \text{argmin}_y \sum_{i=1}^{n} (x_i - y)^2$	Naive Mapping	**0,67787**	**0,15036**	0,14783	0,01312	0,01080
	Depth Buffer Mapping	**0,66193**	0,13732	**0,16330**	0,02067	0,01676
	Occlusion Mapping	**0,63223**	0,10407	**0,19052**	0,03972	0,03344
$g_3 : \text{argmin}_y \sum_{i=1}^{n} \|x_i - y\|^3$	Naive Mapping	**0,57092**	0,14996	**0,25392**	0,01412	0,01106
	Depth Buffer Mapping	**0,56471**	0,13513	**0,26142**	0,02171	0,01701
	Occlusion Mapping	**0,57647**	0,09821	**0,25101**	0,04057	0,03371

parallelizable, so we have developed it on GPU and thus our normal estimation is solved in a few minutes at most for large point clouds.

Once neighbors are retrieved, normal vectors are estimated as an average vector from multiple cross products between those vectors that go from a neighbour to the main point. This basic solution is expected to calculate wrong normal vectors as it does not tolerate noisy neighbourhoods.

The second solution maintains the core of our previous proposal, where a KNN search is performed on GPU. However, we implement an algorithm more robust against noise through a plane fitting methodology. Principal Component Analysis (PCA) allows retrieving the plane that better represents a subset of points (Nurunnabi et al., 2014; Sanchez et al., 2020) through eigenvector decomposition (see Fig. 16). As a result, the calculated normal vectors fit better the expected surface (Fig. 15). The complete algorithm is implemented on GPU as well.

Other libraries are also integrated in our solution to compare their results and performance against our proposed algorithm. Both CPU and multi-core CPU normal estimations from Point Cloud Library (PCL) are integrated. Fig. 17 shows a comparison of the response time of four normal estimation algorithms. The noisy estimation provides a baseline time for a GPU implementation, whereas SVD implements an optimized approach also in GPU. As observed in Fig. 17, our solution scales better with larger point clouds, while it presents a similar response time to multi-core PCL implementation for small point clouds. Therefore, the proposed method is more convenient whether we consider the increasing size of point clouds.

The dot product of the calculated normal vectors, $\hat{n}$, and Y-axis allows to retrieve feasible ground points by defining a threshold, $\hat{n}_{thr}$. Consequently, the complexity of subsequent stages is also reduced. Our methodology is based on the principle of LiDAR returns, where the last impacts are more likely to represent soil points. As a result, the last intersection of each ray is labeled as ground and the canopy is filtered out. The classification of soil and vegetation is approached by assigning volumes to points, as proposed previously. These volumes are here approximated through spheres of radius $R = \frac{GSD}{2}$ (Eq. 11). However, there may be detection errors, as slightly higher elevation points may prevent the detection of surrounding ground points. We avoid most false negatives by defining a tolerance radius, given by $k{\cdot}GSD$. k is ideally expressed as 1, i.e. $k{\cdot}GSD$ is equal to the diameter of spheres.

The method is implemented in GPU so that each point is managed by a different thread. The dot product is first computed to determine the candidate points. Then, n rays are traced from $p_i + (0, y, 0)$ to p_i, provided that n is the number of candidate points and y allows to place the ray origin above the boundaries of the point cloud. Both stages are depicted in Fig. 18. Regarding the use of this procedure for different vegetation fields instead of olive groves, the proposed method is robust enough for detecting feasible ground points. Furthermore, the principle followed to distinguish soil points from the rest of the scenario works for most vegetation environments, as it relies on the detection of the last return of a set of traced rays. However, the method is parameterized through a normal vector threshold and a factor, k, to scale the sphere radius, allowing us to adapt the algorithm to different terrain slopes and point cloud densities.

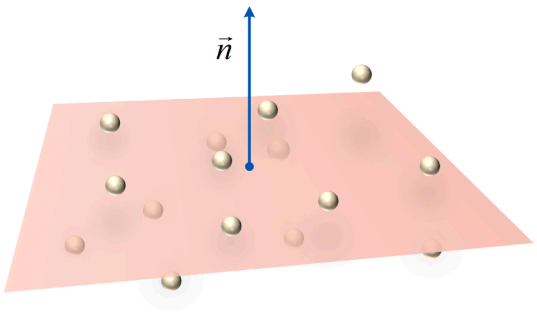

Fig. 15. Schematic representation of normal estimation by detecting the plane that better represents a group of points.

3. Results and discussion

We have evaluated our solution with three scenes of the study area which is following described (depicted in Fig. 19). Furthermore, each scene presents two versions of different point density, depending on the resolution of RGB images during SfM-MVS. Consequently, our input RGB point clouds range from 98 million points to barely 300 thousand points. As reference results, we use point clouds generated by commercial software, e.g. Pix4D or Agisoft Metashape. The quality of the output is measured through three key factors: point density, algorithm performance and similarity of color distribution. We aim to prove the benefits of our solution in comparison with popular software when reconstructing three-dimensional thermal point clouds.

Results from external software were computed using configurations that provided the highest level of detail, both for aligning photos and generating the dense point cloud. Other secondary options not labeled with their accuracy were tested to select the value that guaranteed better results. However, the stage of bundle adjustment was not optimized through the definition of GCPs, as they cannot be marked accurately on TIR images due to the radiance spreading phenomenon. Regarding the stochasticity of SfM-MVS, it is observed that both Pix4Dmapper and Agisoft Metashape software produce non-deterministic outcomes. Consequently, their performance is shown as an arithmetic mean of five results.

All measurements were performed on a PC with Intel Core i7-7700 3.6 GHz, 16 GB RAM, GTX 1070 GPU with 8 GB RAM (Pascal architecture) and Windows 10 OS. The proposed methodology is implemented in C++ along with OpenGL (Open Graphics Library) for rendering. Therefore, parallel algorithms are developed in GLSL (OpenGL Shading Language) through general-purpose compute shaders. Dense RGB point clouds and estimated camera parameters are first computed from Pix4D software and used as the input of our solution.

3.1. Study area

The effectiveness of our proposed methodology was evaluated with UAV imagery captured from a plot located in Mancha Real, Spain, in the region of Jaén. The study area is depicted in Fig. 19 along with the network of GCPs, as well as the geographical demarcation of the region. The surveyed area covers a surface of about 17000 m^2 of an olive grove, although reconstructed areas show a larger area due to the FOV of our sensors. GCPs utilized for the dense RGB point cloud are depicted within the plot. The terrain is characterized by an elevation ranging from 552 m to 572 m. Therefore, our algorithm is exposed to a complex environment, where coacquired images do not present recurrent alignment matrices, partially as a result of non-uniform elevation of the area. Furthermore, some regions of the olive orchard are affected by the pathogen *Xylella fastidiosa*, whose symptoms can be revealed through thermal imagery (Zarco-Tejada et al., 2018). Although this study is not focused on the detection task, it proves the relevance of estimating properly a thermal point cloud.

3.2. Thermal point cloud reconstruction and performance

To evaluate the performance of our methodology, we built an RGB point cloud with SfM-MVS using the RGB images of our study area with their original resolution. Therefore, images are not downscaled and averaged. It is a time-consuming task but also guarantees that nearly 100 million points are estimated. Consequently, this point cloud is the input of the described procedure, whereas the comparatives with external software are based solely on thermal images, as some works proposed previously. Furthermore, we evaluate our three solutions: naive (no

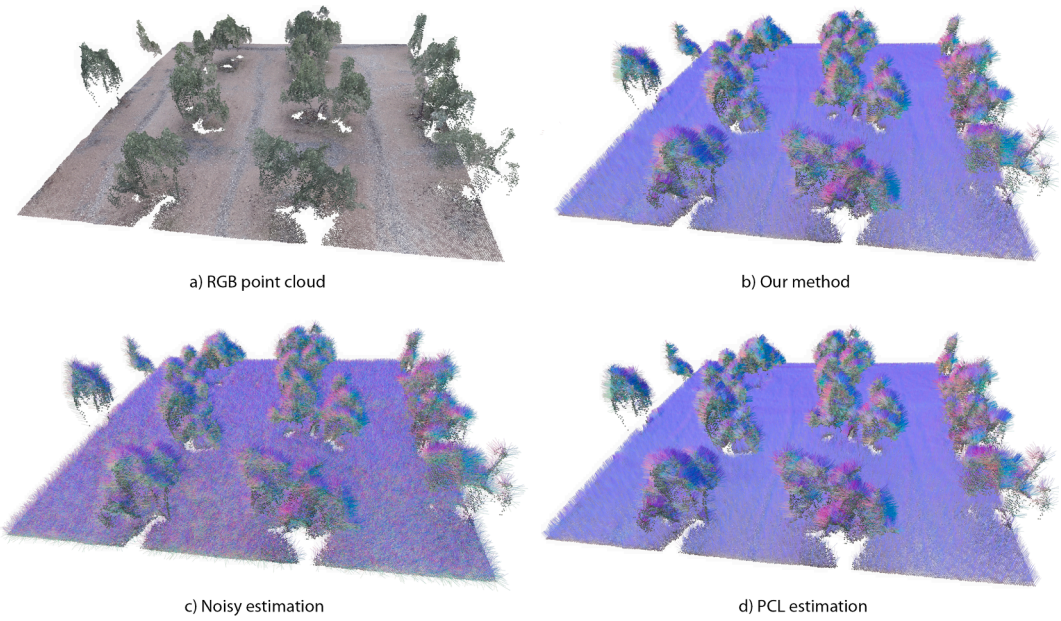

Fig. 16. Comparison of normal estimation results for a point cloud of reduced size. Normal vectors are up-scaled, therefore they are not normalized. Color corresponds to the vector (x, z, y) of each normal. Our method and PCL produces similar outcomes, while ours is optimized for dense point clouds.

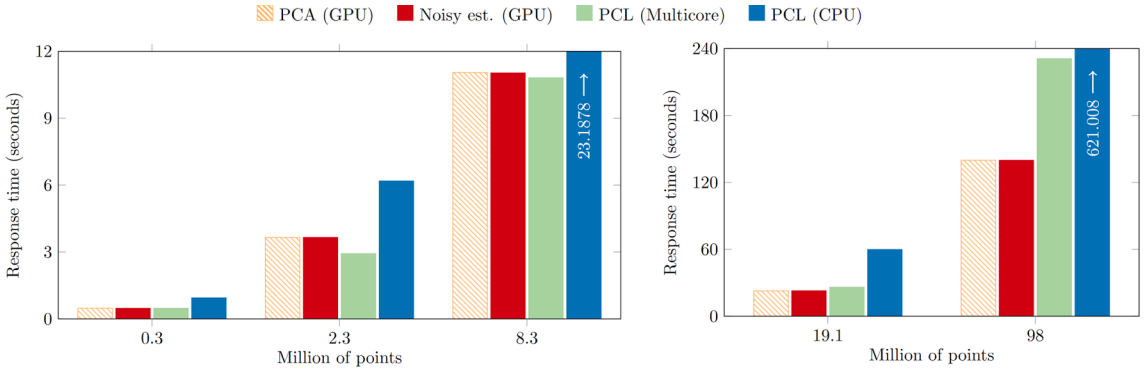

Fig. 17. Response time in seconds for normal estimation algorithms. Both PCL implementations belong to an external library. Results are calculated as an arithmetic mean of five response time samples for each algorithm.

occlusion is considered), depth-buffer (also called visibility test throughout this manuscript) and occlusion (geometrical approach on GPU).

For our methods, we use two different search radius r: $r_1 = 20$ m and $r_2 = 30$ m, centered at $y = \frac{aabb_{min_y} + aabb_{max_y}}{2}$ for every image viewpoint, provided that $aabb_{max_y}$ and $aabb_{min_y}$ are the y coordinate of both maximum and minimum points of the point cloud AABB. The visibility test is configured so that the size of a depth buffer is equal to the size of the original RGB images, i.e. 4000×3000 pixels. BVHs of the occlusion test were constructed using 30-bit Morton codes, ordered through the Radix Sort algorithm, along with a radius of 50 neighbors for collapsing the subtrees. The results are summarized in Table 5. Areas are measured by computing the convex hull that wraps a point cloud through Delaunay triangulation (Shewchuk, 2002), minus the area of internal polygons. The reported response times are retrieved using the arithmetic mean as a fixed aggregation function.

Number of points. From the results, we can see that our thermal point clouds have the largest number of points. Although the number of points by itself does not reflect the quality of the result, these points are expected to be more precise as they are reconstructed from images of higher resolution (RGB) supported by GCPs. On the other hand, point clouds reconstructed solely from thermal data pose several challenges in terms of point density and spatial covering due to the feature extraction phase (Hoegner et al., 2016a; Westfeld et al., 2015; Ham and Golparvar-Fard, 2013a), as reported both in Table 5 and Fig. 22. We lose nearly 16 million points on the naive approach, which are not visible in any thermal image, though the increase of the number of points for r_2 with respect to Agisoft Metashape and Pix4Dmapper results are 358,64% and 746,09% respectively. Occlusion-based approaches build point clouds of lesser size as some points are considered to be occluded when considering a radius equivalent to GSD, although they greatly improve the result of external software. Note that using the radius r_2 barely enhances the size retrieved for r_1, while the response time has significantly increased. Therefore, r_1 is preferable to r_2.

Area and point density. Our algorithms cover the plot more uniformly than the compared methods. Agisoft Metashape fails at some boundary regions, while Pix4Dmapper generates areas indeed very similar to our methods but also presents gaps (see Fig. 22). Additionally, the larger number of points provides an advantage when computing the point density of our solutions. Consequently, the highest point density, given by the naive approach, increases the density by 275,95% and 798,69% from Agisoft Metashape and Pix4Dmapper respectively. As a reference, Webster et al. (2018) manages thermal point clouds of density 776 points/m^2, while (Javadnejad et al., 2020) works with thermal

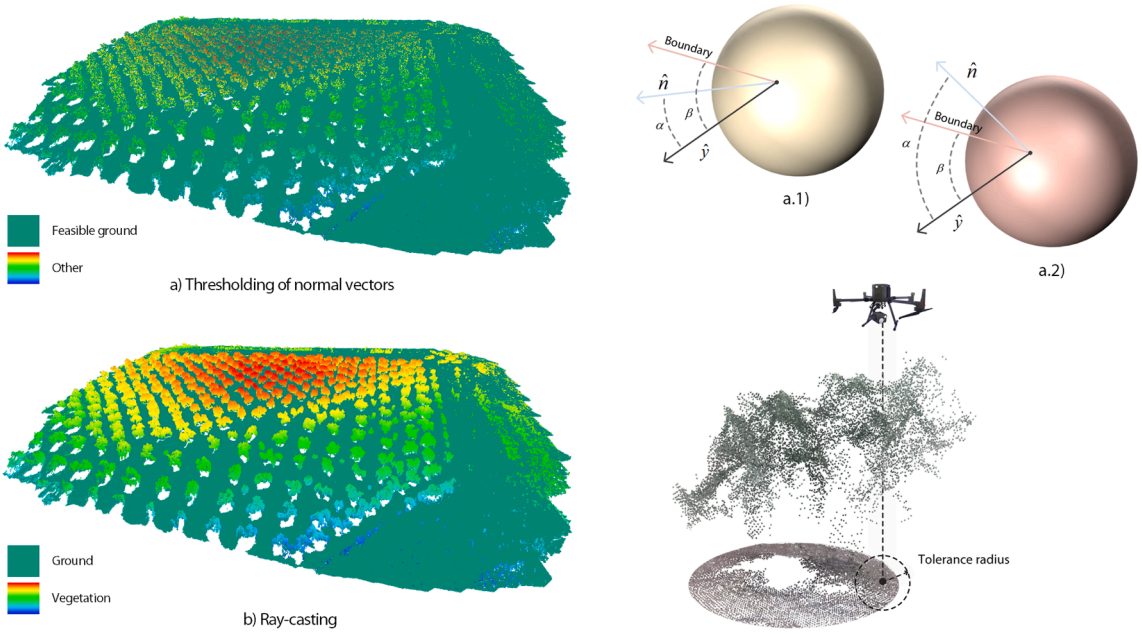

Fig. 18. Classification of ground points with $k = 1$ and a normal threshold of 0,7. a) First step of our methodology; candidate points are displayed as green points. a.1) Valid candidate point ($\alpha \leqslant \beta$), a.2) Point discarded for further processing ($\alpha > \beta$). The angles that both normal and boundary vectors form with respect to the *Y*-axis are defined as α and β respectively. b) Evaluation of ray-casting for each candidate point. Note that less points are rendered with green color at this stage.

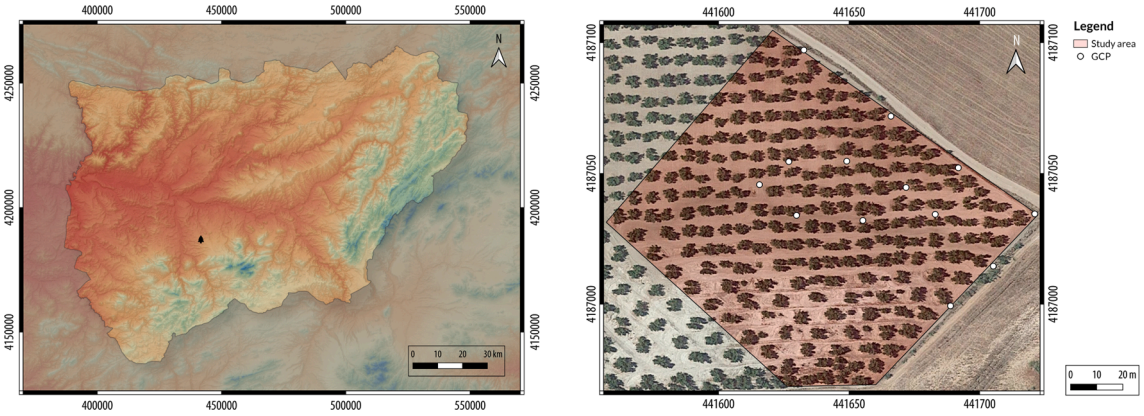

Fig. 19. An overview of the study area. a) Location of the surveyed area in the region of Jaén. b) Study area as a portion of an olive grove. Coordinates are given in WGS84 (EPSG:4326).

datasets of 270 points/m^2 (0,36 ha and 95 images), 336 points/m^2 (2,38 ha and 101 images) and 5.429 points/m^2 (0,35 ha and 165 images). As observed, point density can be improved by increasing the number of coacquired images.

Absolute and normalized response time. The naive approach is the fastest mapping algorithm, whereas occlusion-based methods are also quite competitive, despite the fact that Pix4Dmapper and Agisoft Metashape methods benefit from the use of CUDA-compatible GPUs. As previously described, the naive and depth buffer mapping algorithms are developed as sequential approaches, while occlusion mapping utilizes GPU for solving the occlusion problem. The naive approach offers a minor improvement for the overall response time in comparison with Agisoft Metashape, while normalized response time outperforms commercial software (78,39% and 96,73% less processing time per point with respect to Agisoft Metashape and Pix4Dmapper, respectively). Results of occlusion mapping for r_1 are also remarkable as it builds different BVHs for every image, each one integrating up to 6 million points (r_2 peaks on 14,5 million points).

3.3. Aggregation of thermal data

The proposed methodologies were evaluated in order to prove their accuracy when assigning thermal values to 3D points. However, this chapter is significant whether we assume separate thermal values must be aggregated, despite the fact that they are acquired from different viewing angles. As proposed previously, aggregation functions may be replaced by operators which select the most appropriate value, e.g. assigning priorities based on the viewing angle.

The improvement of applying penalty functions is assessed by measuring the deviation of aggregated values from the observed values in the dataset. For that purpose, we compute the deviation of our thermal point cloud through the Root Mean-Square Deviation (RMSE), Mean Absolute Error (MAE) and standard deviation. However, penalty functions are applied as local optimizers, where the result of an aggregation is chosen as the value that minimizes a distance function (given by the penalty function). This suggests that global measurements are not appropriate for this problem. Average RMSE and MAE (Eqs. 16, 17) are two alternative criteria mainly adopted for unbalanced sets of few items.

Table 5
Performance comparison of the proposed methods. The reported values are averaged over five different executions. r_1 and r_2 correspond to a search radius of 20 m and 30 m respectively. The best results for each measure are highlighted in bold. Mapped images shows the number of thermal images that were successfully mapped into the point cloud. For our method, it shows the number of registered RGB-thermal pairs. For commercial software, images are excluded whether their parameters cannot be estimated or no features are detected.

Attributes	Algorithm			
	Naive Mapping (r_1)	Depth Buffer Mapping (r_1)	Occlusion Mapping (r_1)	Pix4Dmapper
Number of points	82.743.078 points	79.511.469 points	74.181.776 points	9.782.277 points
Area	1,9973 ha	1,9973 ha	1,9973 ha	1,9942 ha
Point density	**4.143 points/m^2**	3.981 points/m^2	3.714 points/m^2	461 points/m^2
Response time	**3 m 36,54s**	7 m 12,18s	4 m 27,83s	13 m 4,4s
Response time per point	**2,617** μs	5,435 μs	3,61 μs	80,185 μs
Mapped images	368 of 410	368 of 410	368 of 410	**371 of 410**
Attributes	Naive Mapping (r_2)	Depth Buffer Mapping (r_2)	Occlusion Mapping (r_2)	Agisoft Metashape
Number of points	**82.767.436 points**	79.511.469 points	74.213.254 points	18.045.885 points
Area	**1,9991 ha**	**1,9991 ha**	**1,9991 ha**	1,7202 ha
Point density	4.140 points/m^2	3.981 points/m^2	3.712 points/m^2	1.102 points/m^2
Response time	7 m 31,01s	12 m 4,38s	9 m 24,92s	3 m 48,31s
Response time per point	5,449 μs	5,435 μs	7,61 μs	12,113 μs
Mapped images	368 of 410	368 of 410	368 of 410	326 of 410

However, it allows quantifying the quality of the adjusted thermal data in terms of distance to image samples.

$$RMSE_{avg} = \frac{\sum_{i=1}^{p} \sqrt{\frac{\sum_{j=1}^{s_i} (S_j - A_i)^2}{s_i}}}{p} \tag{16}$$

$$MAE_{avg} = \frac{\sum_{i=1}^{p} \frac{\sum_{j=1}^{s_i} |S_j - A_i|}{s_i}}{p} \tag{17}$$

where s_i is the number of samples (S_j) from where A_i was calculated. Nevertheless, global RMSE and MAE are also reported in Table 6. Equations of RMSE and MAE are adapted to our set of points as follows (Eqs. 18, 19):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{p} \sum_{j=1}^{s_i} (S_j - A_i)^2}{\sum_{i=1}^{p} s_i}} \tag{18}$$

$$MAE = \frac{\sum_{i=1}^{p} \sum_{j=1}^{s_i} |S_j - A_i|}{\sum_{i=1}^{p} s_i} \tag{19}$$

The average standard deviation of samples for each 3D point is also calculated as expressed in Eq. 20, provided that it does not consider the aggregated value, i.e. it does not vary for different penalty functions.

$$\sigma_{avg} = \frac{\sum_{i=1}^{p} \frac{\sum_{j=1}^{s_i} \left| S_j - \left\{ \overline{S_{i_0}, \ldots, S_{i_{s_i-1}}} \right\} \right|}{s_i}}{p} \tag{20}$$

By the definitions of penalty functions P_1 and P_2 (see Table 6), we can observe that averaged RMSE and MAE are biased towards both functions. Nevertheless, penalty functions were included to minimize error measurements. Indeed, the bias exists and shows that penalty functions reduce the distance from the aggregated value to a set of image samples.

Moreover, methods which take into account the occlusion are expected to yield better results for averaged measures, as the number of samples for each 3D point decreases. However, global RMSE and MAE worsen the results of our penalty-based methods since the minimization was applied for s_i, instead of the overall number of samples, $\sum_{i=1}^{p} s_i$. Nevertheless, aggregated thermal data must be optimized for each point. In that sense, the best value for averaged RMSE is achieved by Occlusion-P_2, whereas Occlusion-P_1 lowers the dispersion measured by the averaged MAE. In any case, the penalty-based methodologies improves the results of the first aggregation row, i.e. approaches with the arithmetic mean as a single aggregation. Penalty function P_1 minimizes the absolute error (MAE_{avg}), while penalty functions P_2 and P_3 penalize higher variations and thus minimizes $RMSE_{avg}$.

Results of aggregating thermal data can also be assessed by computing the distance between histograms of 2D and 3D points. Despite the significant dispersion of image samples, we evaluate if penalty functions produces variations on distance measures. For that purpose, we consider three conventional criteria to compare histograms, such as Pearson Correlation Coefficient (Eq. 21), Hellinger distance (Eq.

Table 6
Measurement of distance from aggregated thermal data, stored at (x,y,z), to 2D thermal samples visible from such point. Blocked samples are omitted by occlusion methods. Errors are computed using intensity values that range from 0 to 1.

Penalty function	Mapping algorithm	Measure				
		Average σ	Average RMSE	RMSE	Average MAE	MAE
No penalty	Naive	0,052	0,054	**0,052**	0,045	0,024
	Visibility	0,049	0,050	0,055	0,042	0,027
	Occlusion	**0,039**	**0,042**	0,054	0,042	0,054
$P_1 : \lvert d(x_i,y)\rvert$	Naive	0,052	0,054	0,053	0,044	**0,024**
	Visibility	0,049	0,051	0,055	0,041	0,027
	Occlusion	**0,039**	**0,042**	0,055	**0,035**	0,034
$P_2 : d(x_i,y)^2$	Naive	0,052	0,053	**0,052**	0,044	**0,024**
	Visibility	0,049	0,050	0,054	0,042	0,027
	Occlusion	**0,039**	**0,041**	0,053	**0,036**	0,035
$P_3 : \lvert d(x_i,y)\rvert^3$	Naive	0,052	0,054	**0.052**	0,045	**0,024**
	Visibility	0,049	0,050	0,054	0,043	0,027
	Occlusion	**0,039**	**0,041**	0.053	**0,037**	0,035

22) and Intersection distance (Eq. 23) (Cha, 2007). We will refer to them as $d_{pearson}$, $d_{hellinger}$ and $d_{intersection}$ for sake of simplicity. $d_{pearson}$ coefficient ranges from −1 to 1, with 1 being a perfect correlation. Zero implies there exist no linear correlation and −1 indicates a perfect negative correlation. On the other hand, $d_{hellinger}$ measures the similarity of two probability functions, where 1 implies that both distributions are orthogonal. The expression of Eq. 22 is simplified if histograms h_1 and h_2 are defined as density functions in [0, 1] ($h_{norm_{1_i}}, h_{norm_{2_i}}$). Finally, the intersection, $d_{intersection}$, is another widely used form of similarity for probability distributions. It returns 1 when $h_{norm_{1_i}}$ and $h_{norm_{2_i}}$ are completely overlapped.

$$d\left(h_1, h_2\right) = \frac{\sum_{i=1}^{n}\left(h_{1_i} - \bar{h}_1\right)\left(h_{2_i} - \bar{h}_2\right)}{\sqrt{\sum_{i=1}^{n}\left(h_{1_i} - \bar{h}_1\right)^2 \sum_{i=1}^{n}\left(h_{2_i} - \bar{h}_2\right)^2}} \tag{21}$$

$$d\left(h_1, h_2\right) = \sqrt{1 - \frac{1}{\sqrt{\bar{h}_1 \bar{h}_2 n^2}} \sum_{i=1}^{n} \sqrt{h_{1_i} h_{2_i}}}$$

$$= \sqrt{1 - \sum_{i=1}^{n} \sqrt{h_{norm_{1_i}} h_{norm_{2_i}}}} \tag{22}$$

$$d\left(h_1, h_2\right) = \sum_{i=1}^{n} min\left(h_{1_i}, h_{2_i}\right) \tag{23}$$

where n is the number of bins of the histograms h_1 and h_2, both of the same size.

Fig. 20 shows the template histogram for the naive approach as well as the histograms calculated from two mapping procedures (without penalty function and using P_3). The rest of approaches present minor changes with respect to this figure and thus are omitted. Therefore, these slight variations are detailed through the aforementioned similarity measures in Fig. 20. It must be noted that $d_{hellinger}$ is transformed into $1 - d_{hellinger}$, so that the three measures can be rendered in the same range. According to the reported results, occlusion-based methods are considered to be more similar to their baseline histogram, specially for the occlusion test. Furthermore, the penalty function P_1, which computes the dispersion as $|x_i - y|$, improves the baseline result for every approach.

3.4. Analysis of thermal point cloud

We have retrieved the temperature for ground and vegetation points once they are recognized. Fig. 21 represents the values of each class from

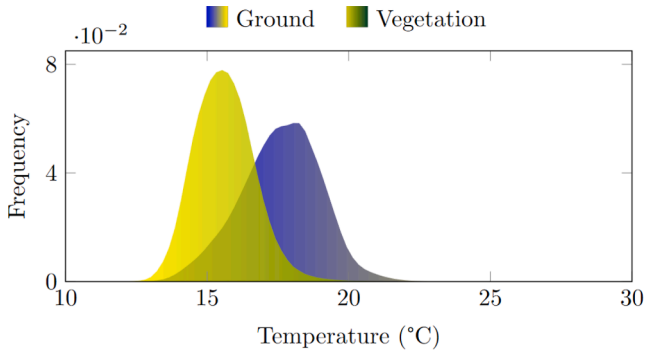

Fig. 21. Frequency function of thermal radiation for three-dimensional ground and vegetation points.

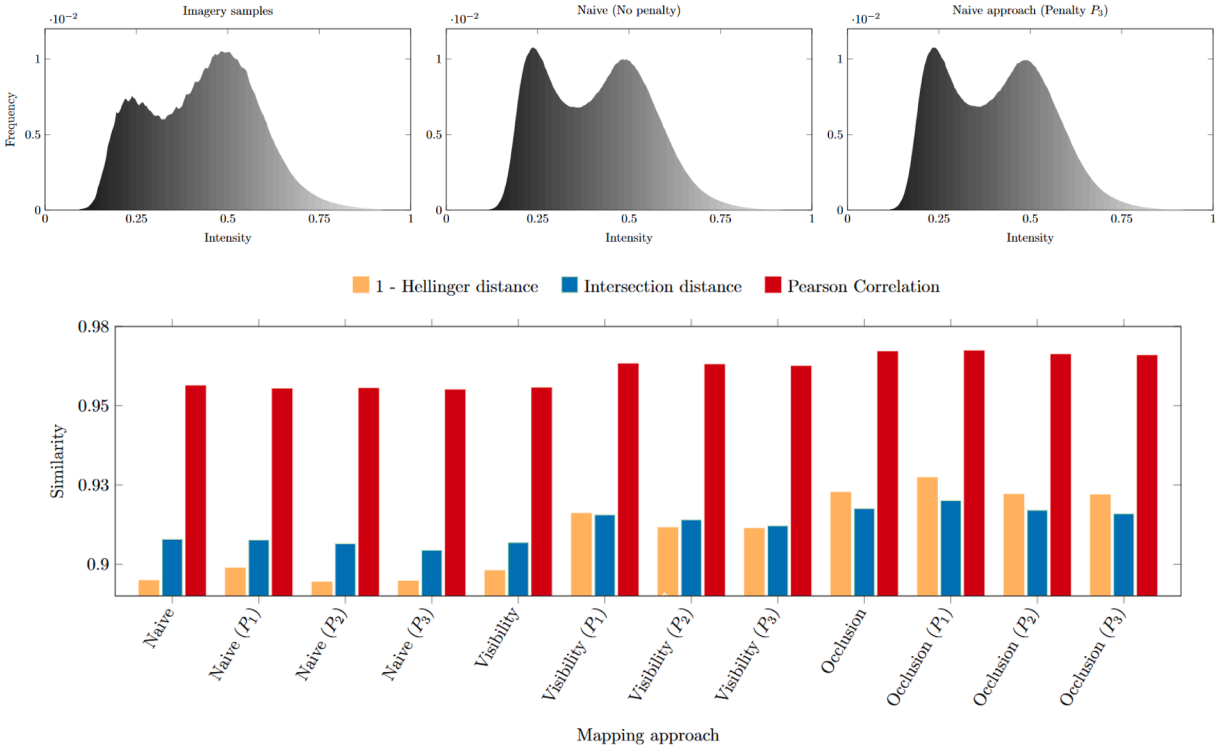

Fig. 20. Intensity distribution for image samples considered while building the thermal point cloud, as well as the distributions recovered after applying two mapping method (naive (no penalty) and naive (penalty-based, P_3)). The density functions of other approaches are omitted as they present similar results by visual inspection. Finally, the similarity measurements between histograms are reported for each configuration, where minor changes in the distribution are accentuated. For sake of simplicity, Hellinger distance is expressed as $1 - d$, so that the three measures can be rendered through the same chart. Hence, the value 1 implies that both histograms are equal ($h_{1_i} = h_{2_i}, \forall i \in [0, n)$), while 0 indicates complete dissimilarity ($h_{1_i} = 1 - h_{2_i}, \forall i \in [0, n)$).

a point cloud of nearly 100 million of points. Accumulated values are presented as a density function based on the point cloud size. Although Fig. 9 shows that vegetation presents overall less emission of thermal radiation, both temperature data-sets are far from being disjoint. It is worth noting that vegetation class also includes dry weeds, which were classified as anomalous hot regions. On the other hand, ground points also cover regions shadowed by trees. However, the local maxima of both thermal distributions clearly yield a relevant value for our two categories. Most thermal values range from 12 °C to 23 °C; anomalous hot regions are detected in vegetation and metal surfaces, whereas cold regions are present at isolated tree and ground points.

3.5. Visualization of the point cloud

To provide better insight into the reconstructed surface, we have rendered the three-dimensional point cloud provided by our solution, as well as the outcome of commercial software. Fig. 22 depicts three results, where the first point cloud is computed through the naive approach by averaging image samples. Agisoft Metashape and Pix4Dmapper results are obtained while measuring their response time. It is worth noting that Pix4Dmapper utilizes a different grayscale distribution, as the software is capable of extracting absolute thermal values from the image dataset. Therefore, such values are normalized considering t_{min} and t_{max}.

As shown in Fig. 22, the result of Pix4Dmapper presents empty regions which correspond to images that were not aligned since relevant key-points were not found. Furthermore, canopies are poorly estimated as they seem much noisier than in the RGB reconstruction, which is known to be correct. Accordingly, trees present higher elevation than in our solution, computed through the RGB point cloud.

Regarding the result of Agisoft Metashape, it presents a wide number of relevant errors. First, canopies are badly estimated, to the extent that they are represented as planar surfaces. Moreover, a significant part of the study area is placed several meters below the rest of the environment. The incorrect reconstruction also leads to noticeable outlier points within the point cloud, although they could be easily filtered out in most cases. Finally, sparsity is also visible for both point clouds, although it is more significant for the Pix4Dmapper result, as reported in Table 3.

Finally, we aim to show the preservation of details acquired in thermal images through Fig. 23. Correction of distortion, as well as accurate projection and aggregation methods, allow us to build a thermal point cloud whose 3D structure is given by an RGB point cloud, whereas no blurring or distortion effect is present.

4. Conclusions and future work

In this work, we described a flexible and automatic algorithm for building dense thermal point clouds. The method receives a dense RGB point cloud as well as a set of RGB and thermal images as input. Coacquired images are then registered so that thermal images can be projected into the RGB point cloud. We also managed to extract temperature values using image metadata and explored the problem of occlusion with two algorithms oriented as 2D and 3D approaches. Hence, we avoid aggregating thermal data from foreground points to background objects. In addition, these methods outperformed commercial software in most of the measured features. The naive CPU approach presented the best overall response time, while the normalized response time was much more competitive for the three methods. Furthermore, the methodology was developed as an accelerated approach to benefit from hardware capabilities. Accordingly, the normal estimation, as well as the most time-consuming algorithms, were developed using GPU. As a result, the reported processing time per 3D point decreased by 78,39% with respect to Agisoft Metashape for building thermal point clouds, though we handle a larger number of points.

In addition, we explored a new algorithm for visualizing thermal

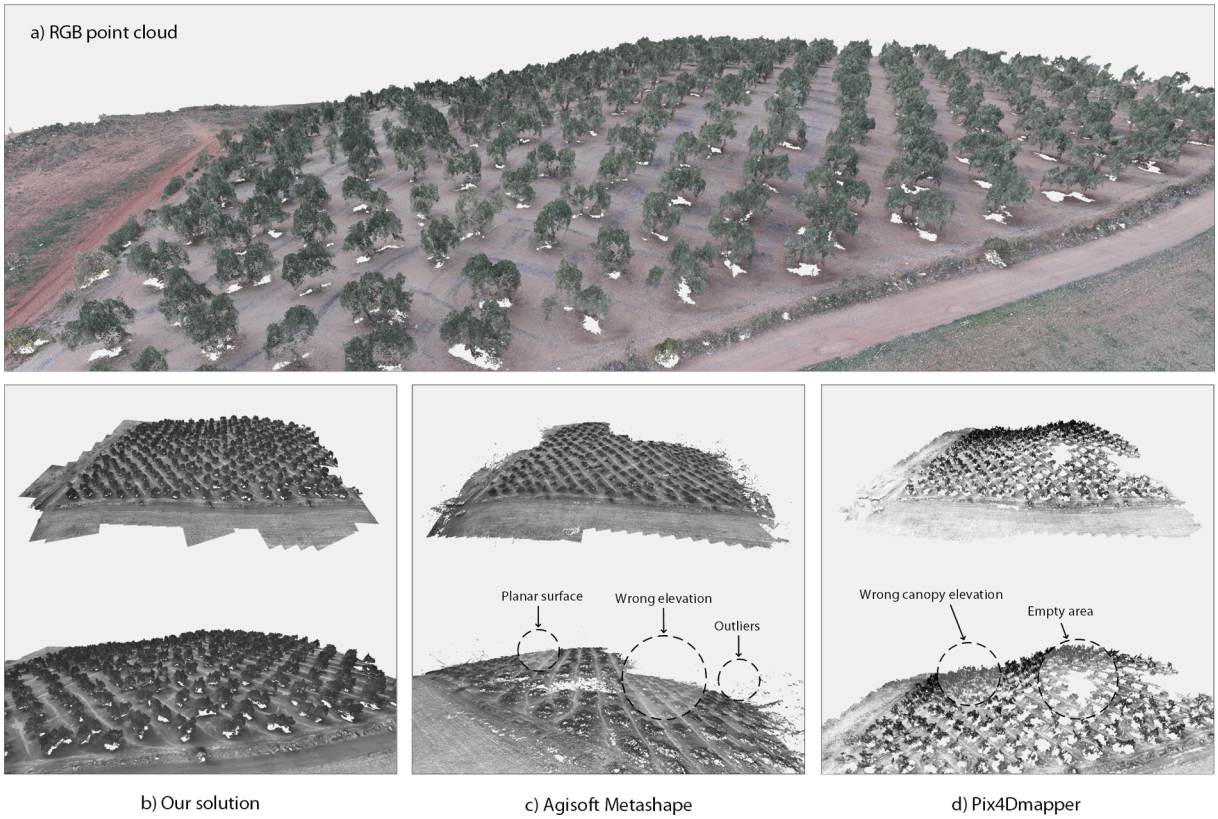

Fig. 22. a) Original RGB point cloud, retrieved with the original resolution, b) thermal point cloud computed using the naive approach, c, d) erroneous Agisoft and Pix4D results, respectively, where their errors are also highlighted.

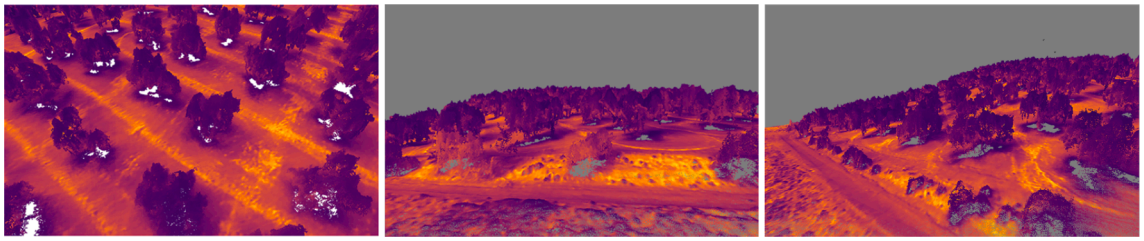

Fig. 23. Close-up representation of a thermal point cloud of nearly 20 million points. As observed, the details of thermal images are preserved and no blurring effects are visible.

points. With it, anomalies were clearly visible. Values close to the outlier boundary were highlighted, whereas intermediate values were mainly represented as RGB points. Although the main objective of this work was to estimate a thermal point cloud, we also provided a methodology to classify and remove ground points, thus focusing on points that are known to belong to vegetation for our study area.

Moreover, we proposed an innovative solution to aggregate multiple values in the remote sensing field. As an alternative for the arithmetic mean, we utilized multiple aggregation operators and each point could select the operator that minimized an error measure. However, the framework was developed so that this procedure could be switched to a single-aggregation approach. Furthermore, we depicted the thermal emission of a point and discussed whether the use of aggregation operators was needed. Finally, we utilized dispersion and histogram distance measures to assess the impact of applying penalty functions. The conducted tests showed that penalty functions clearly reduced the measured error and improved the similarity between the baseline and estimated histogram. However, penalty functions are conceived to minimize a specific error measure and thus any of the described penalty functions are suitable to reduce the distance to TIR radiation samples.

The described methodology is appropriate when SfM-MVS fails due to a lack of key-points, as occurred in our dataset. The density of the estimated point clouds also suggests that this solution would be appropriate for applications where sparsity is a significant drawback. Due to its software capabilities, it is also recommended to minimize the response time as well as for extracting reliable thermal data.

In future work, we would like to conduct a deeper study of the in-situ radiometric calibration, thus allowing us to develop a reliable multi-temporal system based on reconstructed thermal point clouds. Additionally, the registration of RGB and thermal images can be improved to increase the point density of the result. We could also provide even more competitive solutions by developing the complete workflow in GPU. Finally, we aim to carry out further studies about the occlusion problem by comparing several parallel-computing frameworks, and even take advantage of rendering procedures (for instance, the depth buffer step of a rendering pipeline). Regarding the applications of thermal data, we would also like to apply the generated results to control a crop, e.g. by monitoring the presence of pathogens and comparing the detection capabilities with respect to thermal point clouds of lower quality and point density.

The full source code is available at https://github.com/AlfonsoLRz/RGBThermalFusion.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work has been partially supported through the research projects TIN2017-84968-R and RTI2018-099638-B-I00 funded by MCIN/AEI/10.13039/501100011033/ and ERDF funds “A way of doing Europe”, as well as by the Spanish Ministry of Science, Innovation and Universities via a doctoral grant to the first author (FPU19/00100).

References

Alfredo Osornio-Rios, R., Antonino-Daviu, J.A., de Jesus Romero-Troncoso, R., 2019. Recent industrial applications of infrared thermography: A review. IEEE Trans. Industr. Inf. 15, 615–625. https://doi.org/10.1109/TII.2018.2884738.

Behley, J., Steinhage, V., Cremers, A.B., 2015. Efficient radius neighbor search in three-dimensional point clouds. In: 2015 IEEE International Conference on Robotics and Automation (ICRA), pp. 3625–3630. doi:10.1109/ICRA.2015.7139702.

Bustince, H., Beliakov, G., Dimuro, G.P., Bedregal, B., Mesiar, R., 2017a. On the definition of penalty functions in data aggregation. Fuzzy Sets Syst. 323, 1–18. https://doi.org/10.1016/j.fss.2016.09.011.

Bustince, H., Fernandez, J., Burillo, P., 2017b. Penalty function in optimization problems: A review of recent developments. In: Soft Computing Based Optimization and Decision Models. Springer International Publishing, pp. 275–287. doi:10.1007/978-3-319-64286-4_17.

Cha, S.H., 2007. Comprehensive survey on distance/similarity measures between probability density functions. Int. J. Math. Model. Meth. Appl. Sci. 1.

Cohen-Steiner, D., Da, F., 2004. A greedy delaunay-based surface reconstruction algorithm. Vis. Comput. 20, 4–16. https://doi.org/10.1007/s00371-003-0217-z.

Comba, L., Biglia, A., Aimonino, D.R., Barge, P., Tortia, C., Gay, P., 2019. 2d and 3d data fusion for crop monitoring in precision agriculture. In: 2019 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor), pp. 62–67. doi:10.1109/MetroAgriFor.2019.8909219.

Comba, L., Biglia, A., Ricauda Aimonino, D., Gay, P., 2018. Unsupervised detection of vineyards by 3d point-cloud uav photogrammetry for precision agriculture. Comput. Electron. Agric. 155, 84–95. https://doi.org/10.1016/j.compag.2018.10.005.

Connor, M., Kumar, P., 2010. Fast construction of k-nearest neighbor graphs for point clouds. IEEE Trans. Visual Comput. Graphics 16, 599–608. https://doi.org/10.1109/TVCG.2010.9.

DJI, 2018. Zenmuse XT2. dl.djicdn.com/downloads/Zenmuse XT 2/Zenmuse_XT_2_User_Manual_v1.0_en_.pdf.

Evangelidis, G.D., Psarakis, E.Z., 2008. Parametric image alignment using enhanced correlation coefficient maximization. IEEE Trans. Pattern Anal. Mach. Intell. 30, 1858–1865. https://doi.org/10.1109/TPAMI.2008.113.

Feito, F., Torres, J., Ureña, A., 1995. Orientation, simplicity, and inclusion test for planar polygons. Comput. Graph. 19, 595–600. https://doi.org/10.1016/0097-8493(95)00037-D computer Graphics Art.

Furukawa, Y., Hernández, C., 2015. Multi-View Stereo: A Tutorial. doi:10.1561/0600000052.

Gade, R., Moeslund, T., 2014. Thermal cameras and applications: A survey. Mach. Vis. Appl. 25, 245–262. https://doi.org/10.1007/s00138-013-0570-5.

Gong, Y., Yang, Y., Yang, X., 2018. Three-dimensional reconstruction of the virtual plant branching structure based on terrestrial lidar technologies and l-system. Int. Arch. Photogramm. Remote Sens. Spatial Inform. Sci. XLII-3 403–410. doi: 10.5194/isprs-archives-XLII-3-403-2018.

Grechi, G., Fiorucci, M., Marmoni, G.M., Martino, S., 2021. 3d thermal monitoring of jointed rock masses through infrared thermography and photogrammetry. Remote Sens. 13 https://doi.org/10.3390/rs13050957.

Ham, Y., Golparvar-Fard, M., 2013a. An automated vision-based method for rapid 3d energy performance modeling of existing buildings using thermal and digital imagery. Adv. Eng. Inform. 27, 395–409. https://doi.org/10.1016/j.aei.2013.03.005.

Ham, Y., Golparvar-Fard, M., 2013b. An automated vision-based method for rapid 3d energy performance modeling of existing buildings using thermal and digital imagery. Adv. Eng. Inform. 27, 395–409. https://doi.org/10.1016/j.aei.2013.03.005.

Hoegner, L., Abmayr, T., Tosic, D., Turzer, S., Stilla, U., 2018. Fusion of 3d point clouds with tir images for indoor scene reconstruction. ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-1, 189–194. doi:10.5194/isprs-archives-xlii-1-189-2018.

Hoegner, L., Tuttas, S., Xu, Y., Eder, K., Stilla, U., 2016a. Evaluation of methods for coregistration and fusion of rpas-based 3d point clouds and thermal infrared images. Int. Arch. Photogramm. Remote Sens. Spatial Inform. Sci. XLI-B3 241–246. doi: 10.5194/isprs-archives-XLI-B3-241-2016.

Hoegner, L., Tuttas, S., Xu, Y., Eder, K., Stilla, U., 2016b. Evaluation of methods for coregistration and fusion of rpas-based 3d point clouds and thermal infrared images. Int. Arch. Photogramm. Remote Sens. Spatial Inform. Sci. XLI-B3 241–246. doi:10.5194/isprs-archives-XLI-B3-241-2016.

Hou, Y., Volk, R., Chen, M., Soibelman, L., 2021. Fusing tie points' rgb and thermal information for mapping large areas based on aerial images: A study of fusion performance under different flight configurations and experimental conditions. Autom. Constr. 124, 103554. https://doi.org/10.1016/j.autcon.2021.103554.

Jakob, J., Guthe, M., 2021. Optimizing lbvh-construction and hierarchy-traversal to accelerate knn queries on point clouds using the gpu. Comput. Graph. Forum 40, 124–137. https://doi.org/10.1111/cgf.14177.

Jarzabek-Rychard, M., Lin, D., Maas, H.G., 2020. Supervised detection of façade openings in 3d point clouds with thermal attributes. Remote Sens. 12 https://doi.org/10.3390/rs12030543.

Javadnejad, F., Gillins, D.T., Parrish, C.E., Slocum, R.K., 2020. A photogrammetric approach to fusing natural colour and thermal infrared uas imagery in 3d point cloud generation. Int. J. Remote Sens. 41, 211–237. https://doi.org/10.1080/01431161.2019.1641241.

Jiang, S., Jiang, C., Jiang, W., 2020. Efficient structure from motion for large-scale uav images: A review and a comparison of sfm tools. ISPRS J. Photogramm. Remote Sens. 167, 230–251. https://doi.org/10.1016/j.isprsjprs.2020.04.016.

Jurado-Rodríguez, J.M., Cárdenas-Donoso, J.L., Ogayar-Anguita, C.J., Ortega-Alvarado, L., Feito-Higueruela, F.R., 2020a. Semantic segmentation of natural materials on a point cloud using spatial and multispectral features. Sensors 2244. https://doi.org/10.3390/s20082244.

Jurado-Rodríguez, J.M., Pádua, L., Feito-Higueruela, F.R., Sousa, J.J., 2020b. Automatic grapevine trunk detection on uav-based point cloud. Remote Sens. 2043. https://doi.org/10.3390/rs12183043.

Kylili, A., Fokaides, P.A., Christou, P., Kalogirou, S.A., 2014. Infrared thermography (irt) applications for building diagnostics: A review. Appl. Energy 134, 531–549. https://doi.org/10.1016/j.apenergy.2014.08.005.

Lõrinczy, D., 2017. Thermal analysis in biological and medical applications. J. Therm. Anal. Calorim. 130, 1263–1280. https://doi.org/10.1007/s10973-017-6308-2.

López, A., Jurado, J.M., Ogayar, C.J., Feito, F.R., 2021. A framework for registering uav-based imagery for crop-tracking in precision agriculture. Int. J. Appl. Earth Obs. Geoinf. 97, 102274. https://doi.org/10.1016/j.jag.2020.102274.

Mallon, J., Whelan, P.F., 2004. Precise radial un-distortion of images. In: Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004, vol. 1, pp. 18–21. doi:10.1109/ICPR.2004.1333995.

Martínez-Carricondo, P., Agüera-Vega, F., Carvajal-Ramírez, F., Mesas-Carrascosa, F.J., García-Ferrer, A., Pérez-Porras, F.J., 2018. Assessment of UAV-photogrammetric mapping accuracy based on variation of ground control points. Int. J. Appl. Earth Obs. Geoinf. 72, 1–10. https://doi.org/10.1016/j.jag.2018.05.015.

McManus, C., Tanure, C.B., Peripolli, V., Seixas, L., Fischer, V., Gabbi, A.M., Menegassi, S.R., Stumpf, M.T., Kolling, G.J., Dias, E., Costa, J.B.G., 2016. Infrared thermography in animal production: An overview. Comput. Electron. Agric. 123, 10–16. https://doi.org/10.1016/j.compag.2016.01.027.

Meister, D., Bittner, J., 2018. Parallel locally-ordered clustering for bounding volume hierarchy construction. IEEE Trans. Visual Comput. Graphics 24, 1345–1353.

Minkina, W., Dudzik, S., 2009. Infrared Thermography: Errors and Uncertainties. Wiley.

Nurunnabi, A., Belton, D., West, G., 2014. Robust statistical approaches for local planar surface fitting in 3d laser scanning data. ISPRS J. Photogramm. Remote Sens. 96, 106–122. https://doi.org/10.1016/j.isprsjprs.2014.07.004.

Paternain, D., Jurio, A., Beliakov, G., 2012. Color image reduction by minimizing penalty functions. In: 2012 IEEE International Conference on Fuzzy Systems. IEEE. pp. 1–7. doi:10.1109/fuzz-ieee.2012.6250794.

Ribeiro-Gomes, K., Hernández-López, D., Ortega, J.F., Ballesteros, R., Poblete, T., Moreno, M.A., 2017. Uncooled thermal camera calibration and optimization of the photogrammetry process for uav applications in agriculture. Sensors 17. https://doi.org/10.3390/s17102173.

Sanchez, J., Denis, F., Coeurjolly, D., Dupont, F., Trassoudaine, L., Checchin, P., 2020. Robust normal vector estimation in 3d point clouds through iterative principal component analysis. ISPRS J. Photogramm. Remote Sens. 163, 18–35. https://doi.org/10.1016/j.isprsjprs.2020.02.018.

Sanz-Ablanedo, E., Chandler, J.H., Rodríguez-Pérez, J.R., Ordóñez, C., 2018. Accuracy of unmanned aerial vehicle (uav) and sfm photogrammetry survey as a function of the number and location of ground control points used. Remote Sens. 10. https://doi.org/10.3390/rs10101606.

Shewchuk, J.R., 2002. Delaunay refinement algorithms for triangular mesh generation. Comput. Geomet. 22, 21–74. https://doi.org/10.1016/S0925-7721(01)00047-5, 16th ACM Symposium on Computational Geometry.

Sledz, A., Unger, J., Heipke, C., 2018. Thermal ir imaging: Image quality and orthophoto generation. Int. Arch. Photogramm. Remote Sens. Spatial Inform. Sci. XLII-1 413–420. doi:10.5194/isprs-archives-XLII-1-413-2018.

Teza, G., Pesci, A., 2019. Evaluation of the temperature pattern of a complex body from thermal imaging and 3d information: A method and its matlab implementation. Infrared Phys. Technol. 96, 228–237. https://doi.org/10.1016/j.infrared.2018.11.029.

Tsouros, D.C., Bibi, S., Sarigiannidis, P.G., 2019. A review on uav-based applications for precision agriculture. Information 10. https://doi.org/10.3390/info10110349.

Vidas, S., Moghadam, P., Sridharan, S., 2015. Real-time mobile 3d temperature mapping. IEEE Sens. J. 15, 1145–1152. https://doi.org/10.1109/JSEN.2014.2360709.

de Villiers, J.P., Leuschner, F.W., Geldenhuys, R., 2008. Centi-pixel accurate real-time inverse distortion correction. In: Wen, J.T., Hodko, D., Otani, Y., Kofman, J., Kaynak, O. (Eds.), Optomechatronic Technologies 2008, International Society for Optics and Photonics. SPIE. pp. 320–327. doi:10.1117/12.804771.

Vollmer, M., Möllmann, K., 2017. Advanced Methods in IR Imaging. John Wiley and Sons, Ltd. doi: https://doi.org/10.1002/9783527693306.

Webster, C., Westoby, M., Rutter, N., Jonas, T., 2018. Three-dimensional thermal characterization of forest canopies using uav photogrammetry. Remote Sens. Environ. 209, 835–847. https://doi.org/10.1016/j.rse.2017.09.033. URL: https://www.sciencedirect.com/science/article/pii/S0034425717304455.

Westfeld, P., Mader, D., Maas, H.G., 2015. Generation of tir-attributed 3d point clouds from uav-based thermal imagery. Photogrammetrie - Fernerkundung - Geoinformation 2015, 381–393. https://doi.org/10.1127/1432-8364/2015/0274.

Yager, R.R., 2009. On generalized bonferroni mean operators for multi-criteria aggregation. Int. J. Approx. Reason. 50, 1279–1286. https://doi.org/10.1016/j.ijar.2009.06.004 special Section on Interval/Probabilistic Uncertainty.

Yandun Narvaez, F., Reina, G., Torres-Torriti, M., Kantor, G., Cheein, F.A., 2017. A survey of ranging and imaging techniques for precision agriculture phenotyping. IEEE/ASME Trans. Mechatron. 22, 2428–2439. https://doi.org/10.1109/TMECH.2017.2760866.

Zarco-Tejada, P.J., Camino, C., Beck, P.S.A., Calderon, R., Hornero, A., Hernández-Clemente, R., Kattenborn, T., Montes-Borrego, M., Susca, L., Morelli, M., Gonzalez-Dugo, V., North, P.R.J., Landa, B.B., Boscia, D., Saponari, M., Navas-Cortes, J.A., 2018. Previsual symptoms of xylella fastidiosa infection revealed in spectral plant-trait alterations. Nat. Plants 4, 432–439. https://doi.org/10.1038/s41477-018-0189-7.

Zhang, X., Li, H., Dai, M., Ma, W., Quan, L., 2014. Data-driven synthetic modeling of trees. IEEE Trans. Visual Comput. Graphics 20, 1214–1226. https://doi.org/10.1109/TVCG.2014.2316001.

Zhou, S.M., John, R.I., Chiclana, F., Garibaldi, J.M., 2010. On aggregating uncertain information by type-2 OWA operators for soft decision making. Int. J. Intell. Syst. https://doi.org/10.1002/int.20420 n/a–n/a.

Zhu, J., Xu, Y., Ye, Z., Hoegner, L., Stilla, U., 2021. Fusion of urban 3d point clouds with thermal attributes using mls data and tir image sequences. Infrared Phys. Technol. 113, 103622. https://doi.org/10.1016/j.infrared.2020.103622.