

Article

In-Field Estimation of Orange Number and Size by 3D Laser Scanning

Valeriano Méndez ¹, Antonio Pérez-Romero ², Rubén Sola-Guirado ³,
Antonio Miranda-Fuentes ^{3,4}, Francisco Manzano-Agugliaro ^{5,*}, Antonio Zapata-Sierra ⁵ and
Antonio Rodríguez-Lizana ⁴

¹ Department of Applied Mathematics, E.T.S.I.A, Polytechnic University of Madrid, Ciudad Universitaria, Av. Puerta de Hierro, 2, Madrid 28040, Spain; valeriano.mendez@upm.es

² Department of Graphic Engineering, E.T.S.I.A., University of Sevilla, Utrera Road, km 1, 41013 Sevilla, Spain; tao@use.es

³ Department of Rural Engineering, Campus Rabanales, University of Cordoba, Ed. Leonardo da Vinci, Ctra. Nacional IV, km 396, 14014 Córdoba, Spain; ir2sogur@uco.es (R.S.-G.); g62mifua@uco.es (A.M.-F.)

⁴ Department of Aerospace Engineering and Fluids Mechanics, Area of Rural Engineering, E.T.S.I.A., University of Seville, Utrera Road, km. 1, 41013 Seville, Spain; arodriguez2@us.es

⁵ Department of Engineering, ceiA3, University of Almeria, Almeria 04120, Spain; ajzapata@ual.es

* Correspondence: fmanzano@ual.es; Tel.: +34-950-215693

Received: 2 November 2019; Accepted: 11 December 2019; Published: 13 December 2019



Abstract: The estimation of fruit load of an orchard prior to harvest is useful for planning harvest logistics and trading decisions. The manual fruit counting and the determination of the harvesting capacity of the field results are expensive and time-consuming. The automatic counting of fruits and their geometry characterization with 3D LiDAR models can be an interesting alternative. Field research has been conducted in the province of Cordoba (Southern Spain) on 24 ‘Salustiana’ variety orange trees—*Citrus sinensis* (L.) Osbeck—(12 were pruned and 12 unpruned). Harvest size and the number of each fruit were registered. Likewise, the unitary weight of the fruits and their diameter were determined ($N = 160$). The orange trees were also modelled with 3D LiDAR with colour capture for their subsequent segmentation and fruit detection by using a K-means algorithm. In the case of pruned trees, a significant regression was obtained between the real and modelled fruit number ($R^2 = 0.63$, $p = 0.01$). The opposite case occurred in the unpruned ones ($p = 0.18$) due to a leaf occlusion problem. The mean diameters proportioned by the algorithm (72.15 ± 22.62 mm) did not present significant differences ($p = 0.35$) with the ones measured on fruits (72.68 ± 5.728 mm). Even though the use of 3D LiDAR scans is time-consuming, the harvest size estimation obtained in this research is very accurate.

Keywords: orange tree; fruit recognition; K-means; LiDAR; HDS; GNSS; yield estimation; in-field

1. Introduction

The estimation of a crop in its different growth stages is essential when making decisions about harvest, storage, transport, and marketing. In the case of fruit, this estimation is commonly based on manual counts, but they are time-consuming, expensive, and come with huge errors. Therefore, it is necessary to search for automatic counting alternatives, and it is an option to count from information obtained with a camera or a sensor such as LiDAR. Machine vision systems are crucial for automatic fruit detection, the challenges of which were suggested by Sarig [1], as it is a more intuitive approach. The fruit detection must be able to be carried out under different environmental conditions and with the restrictions of shade by leaves, branches, and other immature fruits.

In the literature, diverse works of detection in different types of fruits or harvest can be found, such as almond [2], apple [3–7], cherryfruit [8], cucumber [9], mango [10,11], orange [12,13], pineapple [14,15], or tomato [16].

Fruit detection requires segmentation, shape selection, and identification phases [17]. Segmentation consists of filtering through a colour threshold of the components of the scene that can be considered fruit [18]. Different characteristics of perimeter, area, or compaction allow to select the shapes (blobs) and to identify the fruits one by one.

In bidimensional models, circles can be detected by the Hough transform [19] or by adjusting circular contours [20]. The colour cameras allowed Harrell et al. [12] to implement a robotic orange harvesting system and Grasso and Recce [21] to perform RGB segmentation. Qureshi et al. [11] used clustering of K-nearest neighbour pixels. Qureshi et al. [22] presented a texture-based method for shape recognition using an over-segmentation of super-pixels from the local gradient calculation.

Colour segmentation using charge coupled device cameras returns pixels with RGB graduations or colour composition by addition of the primary colours red, green, and blue, allowing rapid detection of ripe fruit [23]. However, its drawback is its false positives. In addition, shape detection has a high computational cost. Therefore, a suitable alternative is the use of a colour filter followed by shape detection to avoid these false positives.

The colour and shape characteristics allow us to approach its count using 2D photos, filtering by colour or chromaticity, delimiting shapes by contour, and being able to adjust circles. All this allows us to count the total number of fruits and even estimate their size, a major aspect when assessing yield [24].

Regarding problems in fruit detection, they can be classified into illumination and shading by leaves and branches [25]. The fruit illumination is never uniform and depends on factors such as environmental conditions and the position of the fruit within the tree. The detection of shapes, on the contrary, is very affected by the shading, mainly of the leaves, branches, or other fruits. Citrus fruits are trees with high foliar density [26], so 70%–80% of fruits have half or more of their surface darkened. Plebe and Grasso [27] used a system of stereoscopic cameras assembled on telescopic robotic arms to detect a spherical shape. The local detection of the spherical shape from the telescopic arms reduces the occlusion of the sphere. A regression of the spherical object or simply separation of blobs can be achieved depending on the quality of the segmented point cloud of the fruits. Using a LiDAR sensor to perform a 3D scan of a tree could reduce the problem of shading by constructing a point cloud of the tree that was provided of depth. However, the richness of canopy may continue to restrict the complete observation of the tree fruits. Although the LiDAR sensor allows an optimal location, it is large, expensive, and time-consuming [28].

LiDAR technology from a cloud of points allows to reconstruct the detail of a tree and, therefore, measure its elements. However, the reconstruction of all the vegetative elements of the plant due to their complexity cannot be addressed with complete accuracy.

In this sense, general measures have been addressed, such as the diameter at breast height (DBH) [29,30]; tree crown determination, particularly relevant for the adaptation of farming machinery [31] or adjustment of hydro-pneumatic spray dose [32–34]; or the woody structure recreation [35–38]. The methods that allow its estimation are essentially triangulations, adjustment by cylinders, and generation of the skeleton or use of voxels. The digital recreation of canopy is not approachable except the estimation of its surface as derived from the reconstruction of the own crown of the tree [39,40], but the detailed count of the number of leaves is still a huge problem nowadays.

However, fruits and other vegetative elements, such as inflorescences, have characteristics that allow the scene to be segmented by means of a colour filter. This shadowing effect significantly reduces the density of points for trying to select identifiable blobs. For this reason, it is possible to consider using a combination of a LiDAR sensor and a colour photographic sensor.

Therefore, the purpose of the present study was to evaluate: first, the capacity of a K-means algorithm for fruit detection based on a 3D model of pruned and unpruned orange trees, considering

the shading problems that may appear in the trees, and second, to assess the harvest forecasting ability based on the algorithm compared to actual field measurements of harvested fruits

2. Materials and Methods

2.1. Orange Trees Plots

The study was conducted on 26 and 28 December 2017 in a private farm of 14 ha of irrigated orange trees at harvest time. The farm is located in Puebla de la Parrilla (Hornachuelos, Cordoba, Spain), see Figure 1A. The studied crop is a fully developed orange tree plantation of the ‘Salustiana’ variety, with a plantation frame of 6×4 m. For the study, a total of 24 trees were selected, divided into two blocks of 6×2 (Figure 1B). The first block corresponds to trees pruned in March 2017, and the second block corresponds to trees not pruned in that year and, therefore, with larger leaf density and larger crown volume (Figure 2a,b).

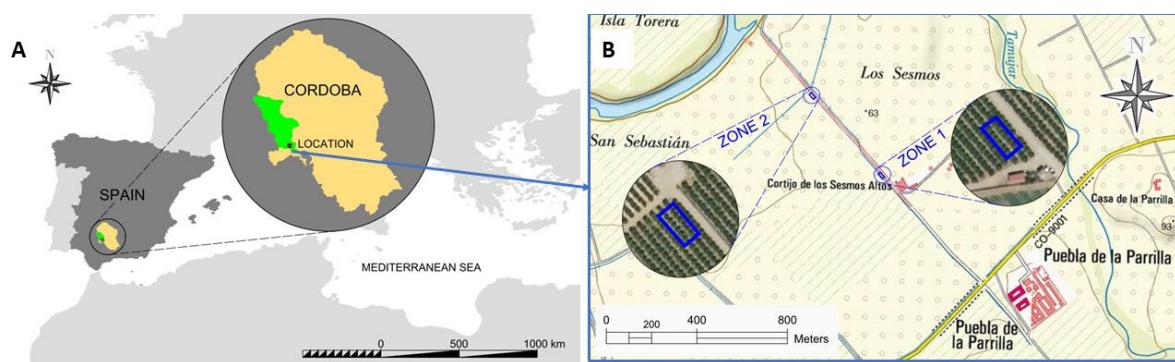


Figure 1. (A) Location of field trials in the south of Spain. (B) Detail of trial plots; Zone 1: 2×6 unpruned trees. Zone 2: 2×6 pruned trees.

The Universal Transverse Mercator coordinates in the central points of the unpruned trial plots (Zone 1 in Figure 1B) are $X = 306,597$ m, $Y = 4,180,494$ m, zone = 30 north and datum ETRS89, with an elevation of 63 m above sea-level. The coordinates of the pruned plot (Zone 2 in Figure 1B) are $X = 306,401$ m and $Y = 4,180,720$ m; the rest of the parameters remain constant. The climate in this area is typically Mediterranean, with three to five hot and dry months (June to September).

2.2. Data

2.2.1. Yield and Orange Sampling

The harvest of each tree was picked separately to obtain the exact number of oranges from each tree individually. The weight of the orange harvest was to be estimated according to the number of oranges detected by the K-means algorithm used (Section 2.3), so a sample of 10 oranges per test tree, selected at different heights and cardinal points, was taken. In each of them, the weight was determined using an electronic weighing scale. Its diameter was also measured with an electronic gauge in the field in order to verify its concordance with the orange size results provided by the algorithm. Finally, the harvest of each tree was individually weighted.



Figure 2. Views of the trial plots. (A) Zone 1, unpruned trees. (B) Zone 2, pruned trees.

2.2.2. Three-Dimensional Modelling Using Laser Scanner

(a) Data Acquisition

A laser scanner, model ScanStation P20 of Leica Geosystems, was used to obtain the point clouds. It is a compact, high-speed pulse scanner with precision, range, and vision of surveying type with an integrated camera that uses Waveform Digitising technology (WFD). These specifications allow 3D position accuracies of 3 mm at 50 m distance on observed objects and the measurement of up to one million points per second.

Prior to the measurements with the scanner, it was necessary to place some checkpoints using 9 steel screws in both zones (see Figure 3). The presence of the control points is essential to make the adjustment of the point clouds taken from each independent position of the laser scanner. The coordinates of these points were made using GPS (Leica 1200) in RTK mode in order to obtain their locations integrated into the geodetic system ETRS89 (European Terrestrial Reference System 1989), the official reference system for Spain in the Iberian Peninsula. According to the manufacturer, of 10 mm + 1 ppm in the horizontal position and 20 mm + 1 ppm in the vertical position, which means a joint error

in three dimensions of approximately 22 mm + 1 ppm. To improve accuracy, 10 measurements were made at each point. The Leica 1200 software allows a basic statistical analysis of the results obtained for each point, deleting those points in which the deviation from the mean had an error of more than 5 mm.

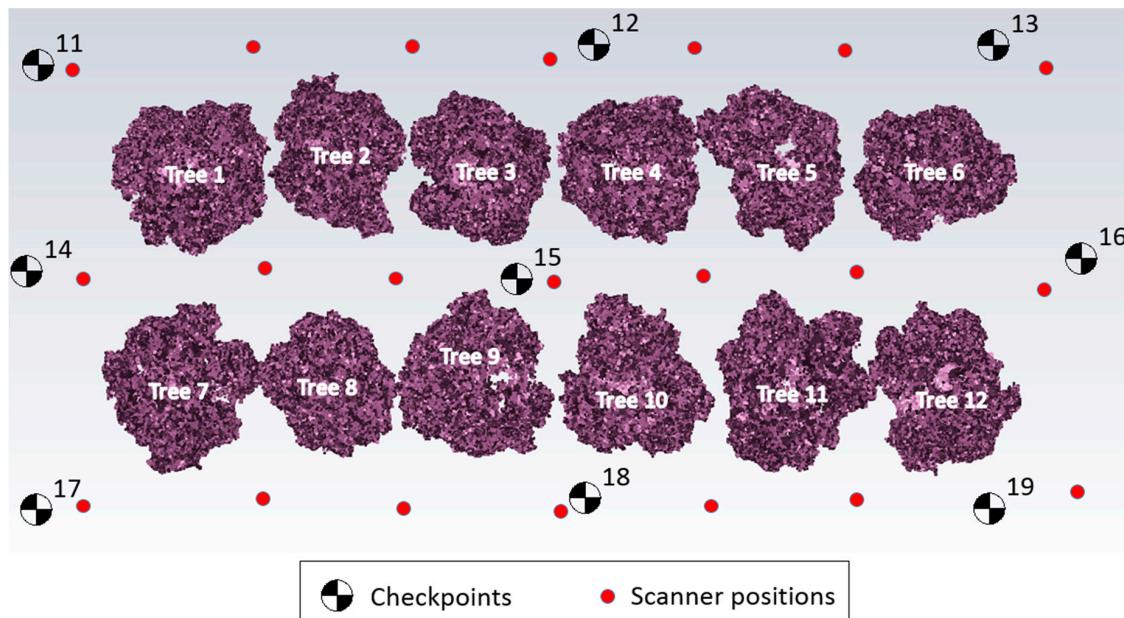


Figure 3. Distribution of checkpoints and scanner positions (Zone 2).

Above the points materialized by the checkpoints, topographical Prism Poles with HDS (high definition surveying) targets were subsequently placed (see Figure 4). These points are intended to be measured from each position occupied by the 3D Laser Scanner. In this way, it is possible to integrate all the scans made and to obtain the correct georeferencing of the entire survey.

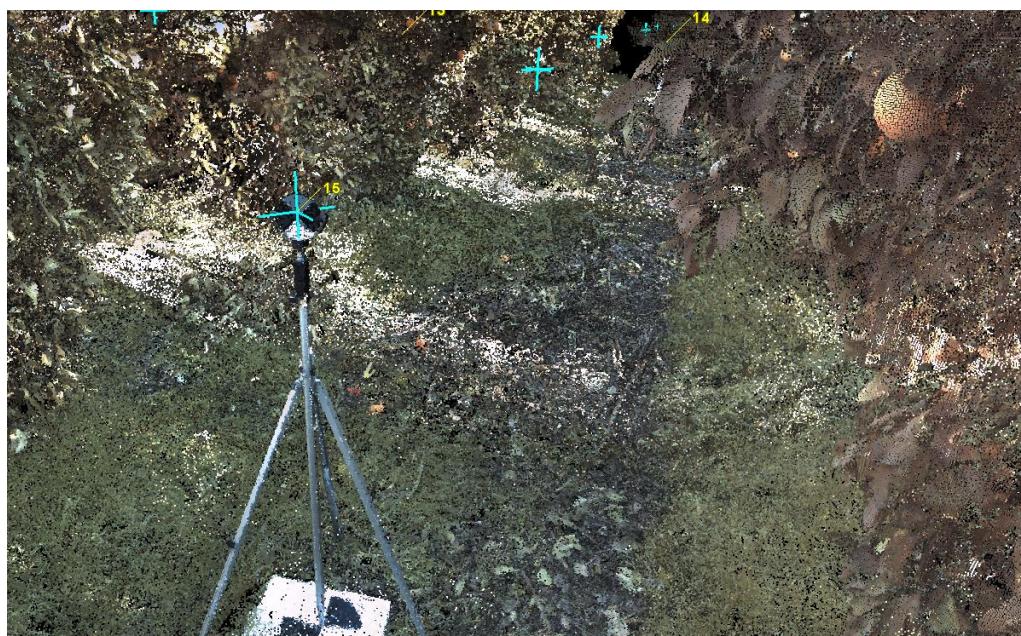


Figure 4. High definition surveying (HDS) targets on topographical Prism Poles.

To ensure accuracy, it is necessary to measure at least two of these checkpoints from each position, but it is best to use three, to perform more robust adjustments. For this reason, the checkpoints were placed in groups of three, in the central area of the crop streets, as can be seen in Figure 3. In order to cover all the trees as optimally as possible, the scanner had to be strategically positioned (Figure 5). One of the main drawbacks of the laser scanner-based HDS measurement technique is that of the “shadow” areas caused by occlusions that prevent complete surveying of the entire surface from a single position, forcing repositioning and new scans to achieve optimal coverage [41]. In this case, in order to cover each tree, and given the environmental conditions, it was required to use four different positions of the scanner, as shown in Figure 5.

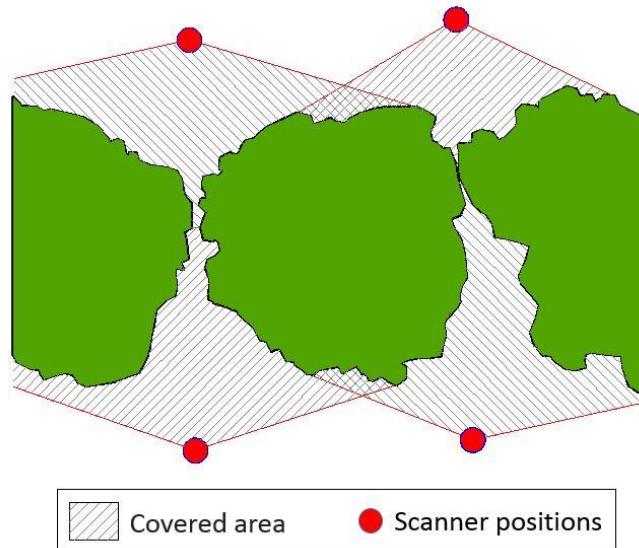


Figure 5. Example of scanner positions to cover the top of a tree.

In a 3D laser scan survey, the density of the obtained point cloud and, consequently, its resolution, is dependent on the scanning distance and angle [42]. In this study, considering the distribution of the scanner positions, the plantation framework and the size of the trees, the distances to the measurement objects were small (between 1 and 3 m).

Then, looking for a proper compromise between the resolution required for the work and the time available to complete it in the field, it was necessary to configure the area to be measured from each of them, especially in the parameters of the scan resolution and image taking. Covered angles scanning between 150 and 170 sexagesimal degrees were in the positions of the outer streets of the crop, but the central street was chosen to perform full circle scans to cover the trees on both sides of the axis of the street from one position.

Note that images are not necessary to obtain point clouds, but by allowing them to be assigned the colours of reality captured by storing RGB values, they make it possible to identify differentiated elements [43], which is especially useful in this work, since ripe fruits (oranges) are clearly differentiated from the rest of the tree by colour. The scanning parameters are shown in Figure 6. The scans made from the outside streets had an average time of 5 min, while in the central street, working with full circle sweeps, the times were 13.72 min.

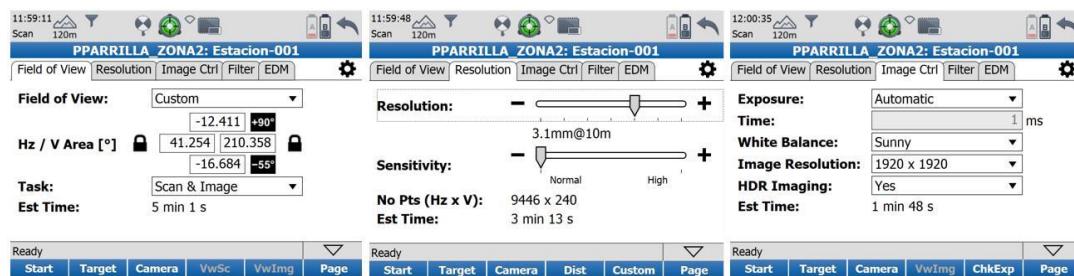


Figure 6. Scanning parameters.

Once the scan was made in a certain position, the equipment moved to a new position, levelling the equipment and measuring the control points placed in the control points. This is necessary for the connection between each of the obtained point clouds, consuming an additional average time of 15 min. Thus, measuring each zone took 8 h.

(b) Data Processing

As the trees were scanned from various angles, from each position of the laser scanner, three of the previously established control points were measured, which allowed the software used (Cyclone) to be able to move and rotate the clouds of points obtained independently in the field so that they can be assembled in the common coordinate system, materialized by the control points. The connection between the point clouds was made using Cyclone software (Leica Geosystems, St. Gallen, Switzerland), achieving internal adjustments between these clouds with maximum errors of 1 mm and adjustments with the coordinates obtained for the control points (measured with GPS techniques) with errors ranging from 5 to 9 mm. The point clouds were exported in PTS format, which conforms to the following structure: the first column corresponds to the X coordinate, the second to the Y coordinate, the third to the Z coordinate, the fourth to the reflectivity and the last three correspond to the values of red, green, and blue (RGB). Figure 7 shows the point cloud generated for one of the scanned trees, and an enlarged detail showing some oranges.

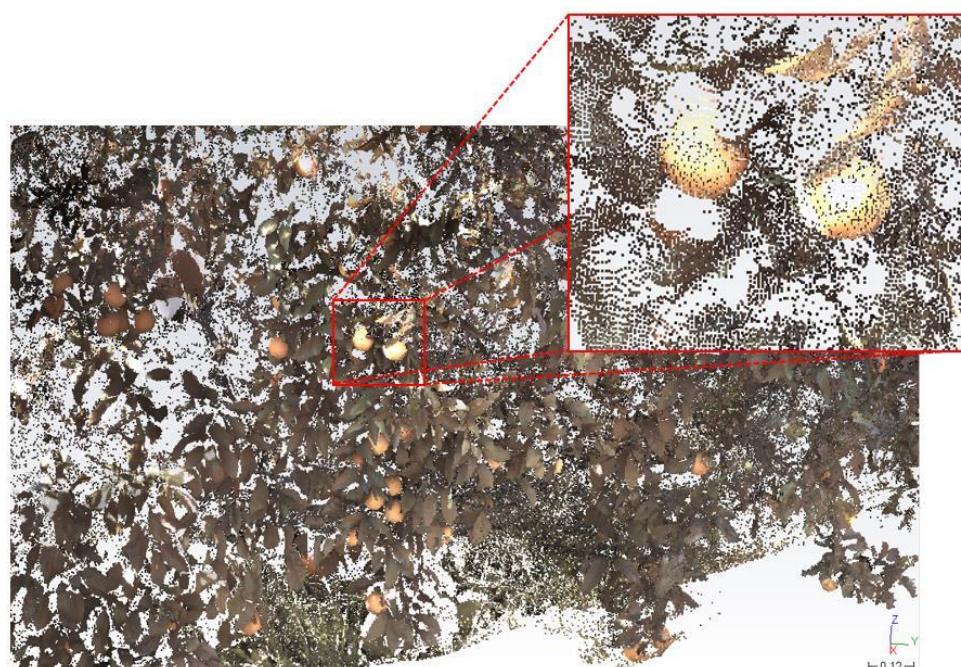


Figure 7. Partial view of one of the point clouds corresponding to an orange tree.

2.3. K-Means Algorithm Application

2.3.1. Data Segmentation

The full reconstruction of the 3D point cloud obtained with the LiDAR sensor is voluminous and extremely slow [28]. The available methods allow the reconstruction of the woody structure or at the local level of other structures, such as leaves or fruits. However, a joint reconstruction of a fruit tree is not usually suitable because the combined mass of leaves, branches, fruits, and flowers does not allow to group and identify each of the elements. Therefore, it was decided to make a colour segmentation based on a direct filter by RGB ($R \geq 180$, $G \geq 50$, $B \leq 105$) or according to the red chromaticity (r) (Equation (1))_ for those values with $r \geq 0.5$ [25].

$$r = R / (R + G + B) \quad (1)$$

2.3.2. Algorithm

Segmentation determines a new cloud of points belonging to interceptions of tree fruits. Obturation problems prevent complete hemispheres from being detected, which can make it difficult to identify oranges. Instead, spherical shapes are obtained, which, in general, constitute a set of blobs that must be individually selected and subsequently identified.

The selection is made according to an ISODATA process [44], which is a K-means where the number of groups or forms are not a priori defined but change during the process. Ridler and Calvard [45] use the ISODATA algorithm to segment images by grey level; Unay and Gosselin [46] or Shruthi [47] identified apples with that algorithm and using 2D colour images.

The following steps apply to the process:

Generate a KD-Tree to facilitate point searches.

Generate a neighbourhood matrix with $r_n < 20$ mm that is supported by the previous KD-Tree, where r_n is the maximum neighbourhood distance.

Initial generation of blobs grouping the neighbouring points to each other.

Readjustment of shapes, merging or dividing the initial shapes by introducing the minimum or maximum dimensions of a fruit ($r_{\min} = 30$ mm and $r_{\max} = 45$ mm, respectively).

The point distribution of the shapes is balanced by applying the K-means algorithm [48]. The objective is that the points are distributed among the different defined shapes minimizing the distances to the centre of gravity.

Eliminate those shapes that have a number of points less than a given number ($n_s < 9$).

The KD-Tree splits the point cloud with planes perpendicular to the coordinate axes so that half the points are on each side of the divisor plane. If “k” divisions are made in a d-dimensional point cloud, a d^{k+1} partition will be done, which reduces the computational costs of searches when generating the neighbourhood matrix in step 2. The computational cost of the process of generating a KD-Tree data structure is given by Equation (2).

$$\text{Cost} = O(N \times \log(N)) \quad (2)$$

where N is the total number of points in the initial cloud [49], and O indicates that the computational cost is less than $N \times \log(N)$.

Given two points P_i and P_j of the point cloud, if the Euclidean distance between them is $0 < P_iP_j < r_n$, then the two points are neighbours. It is indicated by doing $M_{ij} = M_{ji} = 1$, M being a matrix of dimension $N \times N$, and taking the value 0 otherwise. In order to reduce memory consumption, an N-dimensional vector is adopted by storing the positions of neighbouring vertices in the position $M_i = "j, ... "$ and in the position $M_j = "I, ... "$.

As an example, $M_2 = "7, 8, 23"$ indicates that vertex 2 has vertexes 7, 8, and 23 as neighbours. Obtaining the neighbourhood matrix requires each point of the cloud to access the rest to check those

that are at a distance less than r_n . Since the cost of accessing a point on the KD-Tree according to Cormen et al. [49] is $O(\log(N))$, the computational cost of obtaining the neighbourhood matrix is $O(N \times \log(N))$.

The generation of the initial blob groups all the neighbouring points to each other, obtaining a first distribution by compaction criterion. The process will go through the N elements of the neighbourhood matrix $M_i = "j,..."$ so that the points j, \dots will be neighbours to each other and form a blob, but it will be necessary to add to the form the neighbours, if any, of M_j .

The readjustments in the process take place in the following stages, iterating as long as some form is affected, that is, as long as an ungrouping or grouping has taken place:

Ungrouping of shapes that exceed the maximum size. Those shapes where the distance between its centre of gravity and the furthest point (outsider) exceeds the maximum distance (45 mm) are ungrouped. The ungrouping process consists of creating a new shape with the outsider point and then balancing points according to the K-means algorithm [48], which, in this case, would be reduced to 2-means.

Grouping of two shapes in which the distance from the outsider point to the centre of gravity is less than the minimum distance (30 mm) and in which the grouping does not exceed the maximum distance (45 mm).

Assuming there are N_b shapes, go through the list of shapes (with an $O(N_b)$ cost) to identify those that have a dimension greater than the maximum. When this condition is met, the group is split with the creation of a new group with the outsider point. A 2-means operation is then performed to minimize the distances to the centres of gravity of the two resulting shapes. The 2-means clustering has the following cost:

$$\text{Cost} = O(N_b \times \log(N_b) \times \log(N/N_b)). \quad (3)$$

As discussed below for step 5, but being a condition-based process, the total cost of the process will remain as:

$$\text{Cost} = O(N_b \times \log(N_b \times \log(N_b) \times \log(N/N_b))), \quad (4)$$

which is under the level $O(N \times \log(N))$.

The grouping runs through the list of shapes, so it starts from an $O(N_b)$ cost and checks those shapes that are nearby to see if they can join. The cost of this last operation is $O(N/N_b)$, which results in an estimated $O(N)$ cost. In step 5, the K-means algorithm is applied to the whole N_b set of existing shapes. Initially, the list of shapes is crossed and in each one of them, those in which the centres are closer, are selected. An $O(N_b \times \log(N_b))$ cost is estimated for this process.

Subsequently, a point checking operation is executed between the two forms. This implies the route of the list of points of one of the forms with an average cost of $O(N/N_b)$. And if a point had to be exchanged from one form to the other, the recalculation of the two centres of gravity with an $O(N/N_b)$ cost is required. It is estimated as $O(\log(N/N_b))$ because it is a cost under condition. Therefore, it would be $O(N_b \times \log(N_b) \times \log(N/N_b))$, which, in general, will be below the $O(N \times \log(N))$ level. Finally, the debugging of forms that do not have significant points have an $O(N_b)$ cost. Therefore, the expected cost of the algorithm is below the $O(N \times \log(N))$ level.

The process, therefore, consists of the following parameters: maximum neighbourhood distance (r_n), minimum distance of the shape (r_{min}), maximum distance of the shape (r_{max}), and minimum number of points of a shape (n_s). N is the number of points obtained after filtering by colour, and N_b is the number of shapes in which the N points of the cloud are grouped. As the space dimension “ d ” is used ($d = 3$) and “ k ” in the kd-tree structure is the number of levels in which the space is partitioned ($k = 2$ is adopted), and “ K ” in K-means indicates the number of groups, so $K = N_b$ in this context.

2.4. Statistical Analysis

To determine the relationship between the number of oranges detected by the algorithm and the real number of oranges, a [linear regression model](#) (Equation (5)) was fitted:

$$RNO_i = \alpha \cdot MNO_i + \varepsilon_i \quad (5)$$

where MNO_i is the number of oranges for sample i according to the algorithm; RNO_i refers to the real number of oranges for sample i ; and ε_i is the random error. A robust elliptic plot (Relplot) was used to detect and study outliers [50]. We used the contrasts of Durbin–Watson and Glesjer to test the autocorrelation of the residues and their heteroscedasticity, respectively.

The field sampling was divided into three groups to fit Equation 5. Group 1 was composed of yearly pruned trees ($n = 8$) (Figure 8a), while group 2 ($n = 10$) was composed of trees pruned 1.7 years ago (Figure 8b), thus having a denser foliage and the fruits being more difficult to detect. The trees scanned after the sunset were discarded after testing the algorithm with its point clouds since the image was mainly black, and the colour filter could not be applied (Figure 8c).

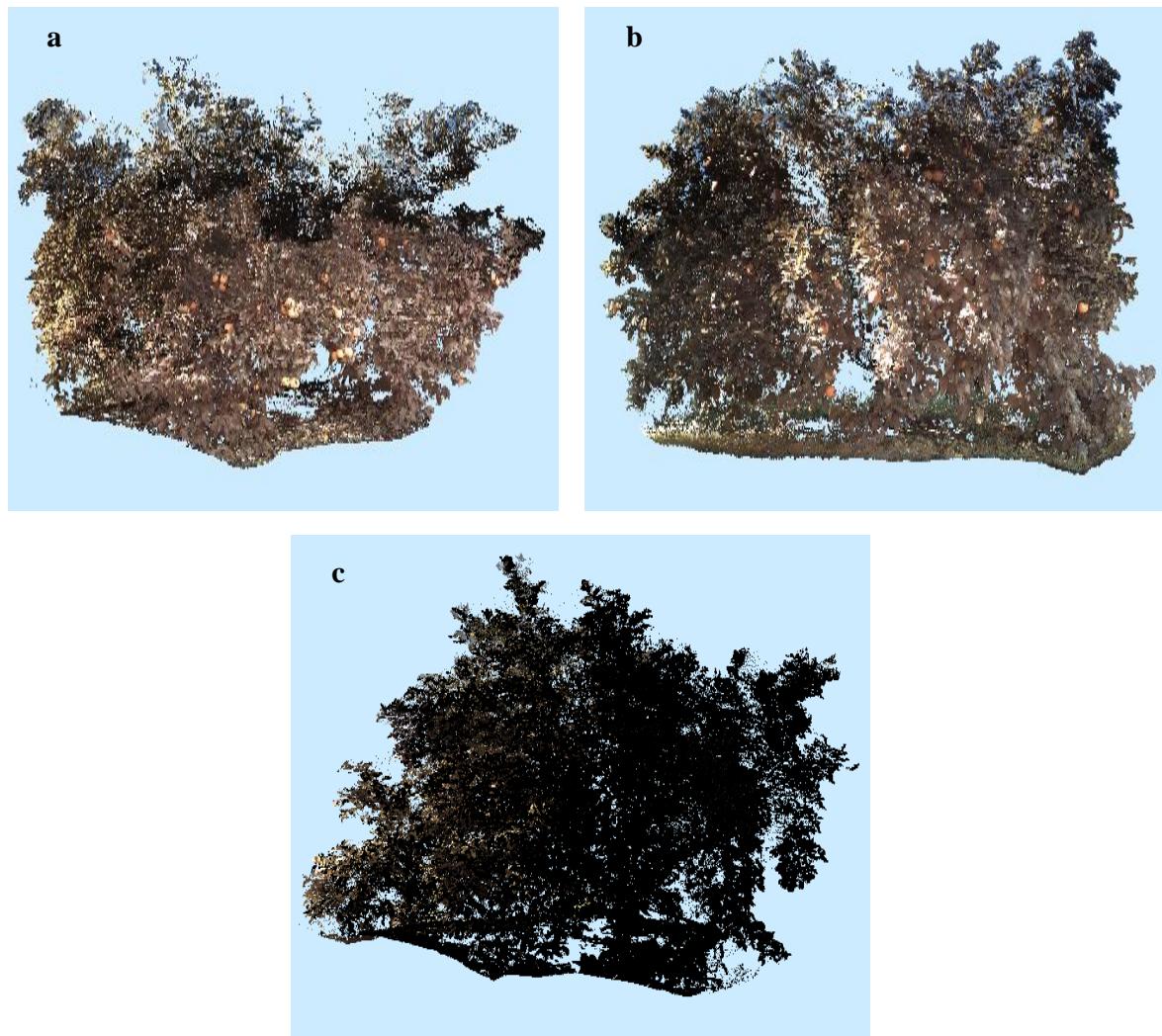


Figure 8. Three-dimensional orange tree rendering prior application to the counting algorithm. (a) Tree pruned, scanned before sunset; (b) tree unpruned, scanned before sunset; (c) tree scanned after sunset, with lack of light.

In the case when the relationship between the field-registered and the algorithm's fruit number resulted as significant (pruned trees), an analysis of variance (ANOVA) was used with a completely randomized design to determine fruit weight differences among trees. Afterwards—for not having significant differences—all data were assembled in the same population, and a harvest size forecast was estimated according to the obtained regression. For this, 2000 simulations were done per single tree, with a varying size depending on the fruit number estimated by the regression between the real fruit number and the one indicated by the K-means algorithm. Each one of these samples presented a normal distribution with a mean value set by the result of the regression in each tree, and a variance was given by the root mean squared error of the performed ANOVA, as an estimator of the population's variance. Thus, 2000 equiprobable harvest sizes were obtained for each tree, with a normal distribution. The sum of these random variables (eight trees) proportioned the harvest estimate.

In relation to the fruit diameter, a t-test for independent samples between the field-measured and algorithm's diameter values was conducted. Finally, a linear regression model between the fruit diameter and weight was obtained to determine the accuracy of the fruit diameter as an estimator of the fruit weight, for being one of the data proportioned by the K-means algorithm.

For descriptive statistical analysis of data and regression analysis, R [51] and SPSS v. 25 were used (IBM, Armonk, NY, USA).

3. Results

3.1. Orange Count with K-Means Algorithm

In both the pruned and unpruned areas, it was necessary to discard some trees for having been scanned after sunset, which made it impossible to use the colour filter at a later stage. The field counting of the fruit per tree was performed after the harvesting in the case of the pruned area ($n = 8$). The average fruit number was between 278 and 470 fruits at each tree. The counting algorithm gave values between 209 and 351 fruits, underestimating the number of fruits in every tree, from 11% to 40%.

Outliers were not detected by the report, so the real and modelled fruit correlation was performed by ordinary minimum squares. A linear correlation with Pearson coefficient $r_{xy} = 0.79$ (p -value = 0.019) was found. The hypothesis of normality and homoscedasticity were not rejected for having Shapiro–Wilk and Glesjer test p -values of 0.55 and 0.73, respectively. The residual standard error was equal to 59 fruits. The relationship between both variables is shown in Figure 9a.

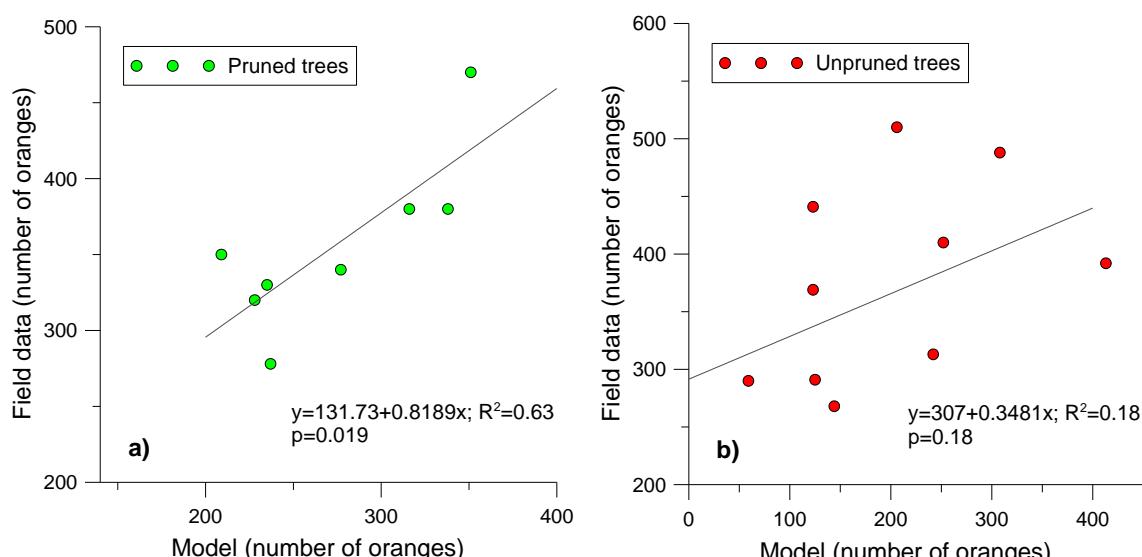


Figure 9. Relationship between the field-registered fruit number and the K-means algorithm's fruit number. (a): Pruned trees; (b) Unpruned trees.

The field counting of the fruit per tree was performed after the harvesting in the case of the unpruned area ($n = 10$). The average fruit number was between 268 and 488 fruits at each tree. The model estimated from 59 to 413 fruits (Figure 9b), with reductions between 22% and 79%, higher than those found in the pruned trees. In this case, outliers were not detected according to replot, estimating the model by ordinary minimum squares, with a coefficient of $r_{xy} = 0.43$ (p -value = 0.18, ns), and a root mean square error of 82 fruits.

According to the results, the algorithm requires the aforementioned colour filtering, as it provided scores between 25 and 67 fruits in the trees scanned after sunset, clearly lower than the real ones. In addition, not implementing the filter dramatically increased the number of points, with the associated increase in the computing time. If this cost is not compensated by the possibility to determine a spherical regression in the fruit areas, the fruit account becomes unaffordable.

Figure 10 shows a graphical output of the algorithm result used in one of the studied trees and the detected fruits. Blobs with point numbers below the minimum threshold were discarded.

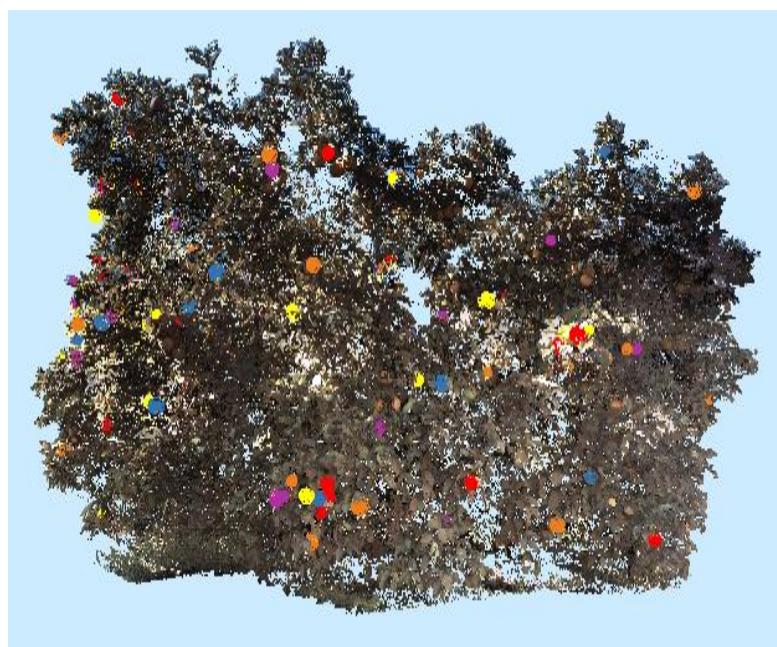


Figure 10. Tree #21 reconstruction from the data registered by the LiDAR sensor and fruit blob reconstruction after the orange point filtering. The fruits have been represented in five different colours to achieve a better visualization, thus avoiding overlapping of oranges.

3.2. Yield Estimation

The ANOVA of the fruit weight of the trees had a p -value equal to 0.31, so the hypothesis of equality of mean weights among trees is not rejected. Normality hypotheses of subgroups were likewise not rejected (Shapiro–Wilk p -values between 0.12 and 0.73), and the same occurred with homoscedasticity (Levène p -value of 0.08). For this reason, it can be assumed that fruits came from the same population regarding the harvest estimation. Fruits presented a mean weight value of 197.53 ± 4.22 g, and a Shapiro–Wilk p -value = 0.09, thus not rejecting normality.

Table 1 shows the estimated harvest according to the usage of direct algorithm and regression found in Figure 9a. It was obtained by multiplying the regression estimated fruit number by their mean weight. These data can be used to achieve a harvest mean forecast, according to the regression model, with very feasible results for the plot (sum of the eight trees), Table 1. The difference between them was 2.5%. As it can be observed, the use of a regression model is necessary.

Table 1. Comparison between real and algorithm-estimated harvest in the pruned trees.

Harvest (kg)	Tree								Total (kg)
	1	2	3	4	5	6	7	8	
Real	78.0	76.0	56.5	59.0	52.0	75.5	86.0	66.0	549
Regression	80.7	82.8	64.0	70.8	64.3	59.8	77.13	62.9	563
Algorithm	64.9	67.4	45.1	53.2	45.5	40.1	60.7	43.8	421

The estimation of the common variance, given by the mean square error, was equal to 1400.86 g². The random sample simulation ($n = 2000$) proportioned a mean harvest estimation of 563 kg, with minimum and maximum values of 557 and 569.3 kg, respectively. The 96% confidence interval equalled 559.7 and 566.8 kg, which allows for its estimation with a low uncertainty value.

One of the most relevant parameters of the quality of oranges refers to the size and, therefore, the diameter of the fruit. The fruit diameter estimated by the algorithm presented a value of 72.15 ± 22.62 mm. Diameter values registered on the field were equal to 72.68 ± 5.728 mm (horizontal plane axle). No significant differences were found between them ($p = 0.35$). It is noticeable that the mean size of the fruits resulted as nearly equal. Even when differences in the standard error are observed because the radius distribution of the algorithm is of heavier tails than the distribution of the samples, it is remarkable that the average size of the oranges is almost the same. The diameter values registered on the field presented small variations with respect to their mean value (variation coefficients of 3.5%, 7.3% and 8.1% for the three main axes).

The fruit weight depends on the fruit diameter for the sampled fruits (Equation (6)). A high correlation between the measured diameters and the fruit weight was observed (coefficient r_{xy} values equal to 0.89 and 0.87 for the horizontal axis, and 0.78 for the Z axle, with $p < 10^{-10}$ in every case). The obtained equations were:

$$\begin{aligned} \text{Fruit weight}_i &= -215.6 + 5.596 \times \text{Diameter}_i + \varepsilon_i^{\text{a, b}} \\ \text{Fruit weight}_i &= -212.8 + 5.344 \times \text{Diameter}_i + \varepsilon_i^{\text{c}} \end{aligned} \quad (6)$$

^{a,b}: perpendicular axis in the horizontal plane; ^c: Z axle.

4. Discussion

4.1. Orange Count with K-Means Algorithm

The time required for data collection in the field is quite long. However, it provides very reliable results and a high resolution, showing small details, which are very relevant for fruit detection. It also provides relevant information such as RGB and reflectivity. All these advantages make it serve as a reference method to compare with other cheaper and faster methods. In fact, this study opens new perspectives for alternatives, such as photogrammetry-SFM (structure from motion) and automatic image detection techniques based on Machine Learning, but always taking, as a reference, the information generated with the 3D laser scanner. In addition, new 3D scanning equipment is already coming to market that is cheaper and faster.

There are two problems that can affect the success of fruit detection: lightning and occlusion. Bulanon [5], Tabb [6], and Kondo [16] found that lightning could be a significant problem by affecting filtering. The occlusion minimizes the visible fruit area and disrupts the affected fruit shape [25]. The main cause for occlusion is the presence of leaves, branches, or other fruits.

The algorithm underestimation of the fruit number results logical, as the 3D scan loses information when getting deeper inside the canopy, mainly due to the occlusions produced by branches, leaves, and fruits found in the outermost part of the tree crown. This generates a data reduction and hides fruits or shows small parts of them unable to be recognized by the algorithm. In this sense, Figure 11 shows a significant example taken from one of the trees. Points were taken from a slide of 20 cm width,

taken at 1.60 m height from the ground, and the top projection shows how the point density decreases when the depth increases. This pattern is repeated in the rest of the sections.

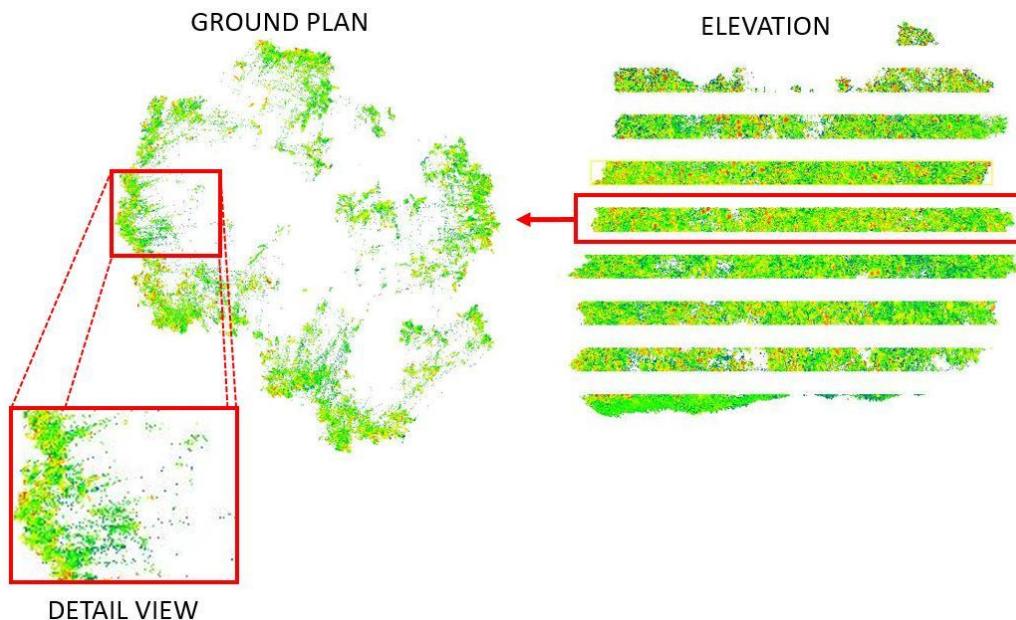


Figure 11. Point density measured in a tree crown. The reduction of this parameter when increasing the sampling depth can be observed (ground plane and detail view).

This limitation is inherent to the 3D scan, and it is also present in the UAVs, as they detect the outermost part of the canopy. However, the correlation found in the pruned trees (Figure 9a) is interesting, as it allows for the establishment of a significant linear trend between real and modelled values.

In the case of unpruned trees, the lack of correlation between real and modelled fruits can be due to an important shading problem [52,53]. Even if occlusion is a natural phenomenon in trees [23], produced by leaves or fruit clusters [13], the higher leaf density of the unpruned trees importantly accentuates this problem. In addition, the difficulty in obtaining full models of the unpruned trees should be pointed out. This was because the leaves sometimes formed a continuous canopy surface, as can be seen in Figure 12 (Zone 1), especially in the northwest row. All these reasons explain the fact that the unpruned trees present a higher mean root square error (82 fruits) than the pruned ones (59 fruits).

In the trees scanned after the sunset, the results obtained could be explained by the colour absence for a preliminary filtering in addition to the shadowing of leaves. As indicated by Jiménez et al. [23], the application of this filter is key for mature fruit detection and computation cost reduction.

Finally, it should be remarked that a clear limiting factor in this type of experimental research is the slow field data acquisition as measuring each study area took a mean time of 8 h. In this sense, counting with faster commercial technology is key to use this technique. Newly-developed laser scanners (e.g., RTC360 by Leica Geosystems), recently available, will allow for an important reduction of about 75% time with respect to the one used in this study. In the near future, new systems will allow for the in-motion point and colour capture (e.g., the scanner BLK2go by Leica Geosystems), reducing time.

Another factor to take into account is the importance of reducing the volume and weight of equipment, which will allow in the near future for the mounting of these devices on UAV platforms. This will increase the data acquisition speed and, therefore, the study area size. LiDAR sensors currently mounted on UAVs capture 3D point clouds. Nevertheless, they cannot register RGB information, which, as shown in the present paper, is mandatory for the automatic fruit detection.

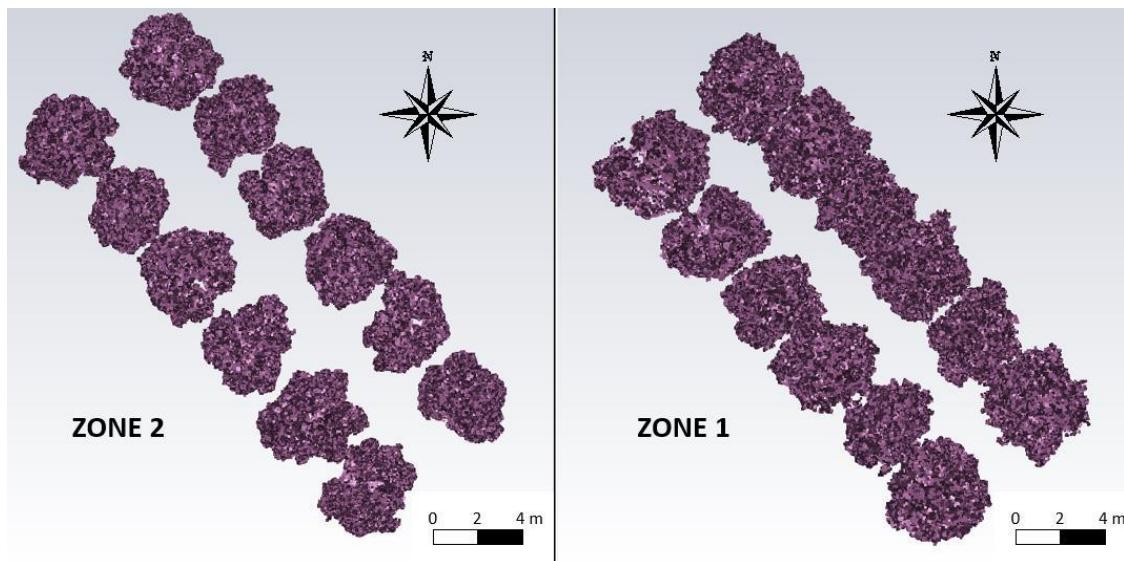


Figure 12. Top view of the trees inside the study field. Area 1: 2×6 unpruned tree block. Area 2: 2×6 pruned tree block.

4.2. Yield Estimation

The use of the proposed algorithm makes it possible, once the regression is obtained (Figure 9a), to accurately estimate the harvest size. The fact that the total harvest simulations were slightly higher than the real ones indicates that the 10 samples taken per tree moderately overestimated the mean weight of the fruits in the whole tree. Anyway, the harvest size estimation is correct, as the overestimation oscillated between 1.4% and 3.6% of the total harvested weight.

In the future, it would be useful to study the leaf density threshold to allow the algorithm to obtain accurate fruit number predictions and, therefore, proper harvest forecasts, as the relationships between modelled and real fruit numbers must be determined.

In the same direction, the good reproduction of the fruit radius by the algorithm enables the harvest estimation, by taking into account its relationship with the fruit weight, as indicated by Sharifi et al. [24] in ‘Thompson’ variety orange fruits, where equations dependent on the main diameters were obtained with very similar coefficients to the ones obtained in the present study, as well as in Tabatabaeefar et al. [54].

The determination of the experimental Equation (6) is interesting as the K-means algorithm indicates the fruit radius, which also makes it possible to estimate the fruit harvest, once the number of fruits and the equation to relate the real and algorithm productions (Figure 5a) are known. However, given that the algorithm gives a sphere diameter, and taking into account that its volume depends on the cube of its radius, the use of the next experimental equation is proposed for the variety comprised in the present study to estimate the fruit harvest:

$$\text{Fruit weight} = 26.729 + 0.000406 \times (\text{Diameter } a \times \text{Diameter } b \times \text{Diameter } c) \quad (7)$$

$$R^2 = 0.88, p < 10^{-10}$$

where dimension “a” is the longest diameter; b is the longest dimension perpendicular to a; and c is the longest dimension perpendicular to “a” and “b”. In this particular case of obtaining a unique dimension from the K-means algorithm, Diameter a = Diameter b = Diameter c would be assumed.

To be of relevance to marketing decisions, data is required weeks before harvest; therefore, these techniques may be very useful in the near future. This is possible for fruit number; however, fruit size changes in a non-linear way before harvest, so a time series of size measurements would be required in order to assist the decision making.

5. Conclusions

The use of 3D scanning and a K-means algorithm proved effective in estimating the crop structure and also the position and the size of the fruits. There is a significant linear correlation between the number of oranges in the pruned trees and those estimated with the K-means algorithm presented. The acquisition of this relationship between both variables is necessary because the direct use of the algorithm does not provide precise results.

The algorithm provides accurate orange diameter estimates, which is of high interest in determining the size of the fruits of the harvest. An experimental linear relationship between the diameter of the fruits and their weight, with high determination coefficient, has been obtained.

The knowledge of this experimental relationship, and of the relationship between the number of real oranges and the number modelled with the algorithm allows us to obtain crop estimates with a small margin of error. Although the use of a 3D laser scanning is a highly time-consuming method nowadays, it opens up new perspectives in the yield assessment of oranges and orange count. Likewise, due to its high precision, it could serve as a reference method to compare with other cheaper and faster methods in the future, such as photogrammetry-SFM (structure from motion) and automatic image detection techniques based on machine learning, or new faster 3D scanning equipment. In summary, the results of this study are interesting and needed as they show potential for uses of 3D laser scanning in the future for yield estimation and, thus, represent valuable basics for further study and development of 3D laser scanning.

Author Contributions: Conceptualization, A.R.-L., V.M.; methodology, V.M. and A.P.-R.; resources: A.P.-R. and F.M.-A., data collection, A.P.-R., R.S.-G., A.M.-F., A.R.-L.; formal analysis: V.M., A.R.-L., A.P.-R., R.S.-G., A.M.-F.; software: V.M., A.P.-R.; writing—original draft, V.M., F.M.-A., A.R.-L., A.P.-R., A.Z.-S.; supervision, A.R.-L.; writing—review and editing, F.M.-A., R.S.-G., V.M., A.Z.-S., A.M.-F.

Funding: This research received no external funding.

Acknowledgments: The authors would like to thank the owners of the orange tree grove for his collaboration.

Conflicts of Interest: The authors declare no conflict of interest. Mention of trade names or commercial products in this publication is solely for the purpose of providing specific information and does not imply recommendation or endorsement by the authors of this article.

References

1. Sarig, Y. Robotics of Fruit Harvesting: A State-of-the-art Review. *J. Agric. Eng. Res.* **1993**, *54*, 265–280. [[CrossRef](#)]
2. Underwood, J.P.; Hung, C.; Whelan, B.; Sukkarieh, S. Mapping almond orchard canopy volume, flowers, fruit and yield using LiDAR and vision sensors. *Comput. Electron. Agric.* **2016**, *130*, 83–96. [[CrossRef](#)]
3. Grand d’Esnon, A. Robotic Harvesting of Apples. In *Proceedings of Agri-Mation 1 Conference and Exposition*; ASAE: St. Joseph, MI, USA, 1985; pp. 210–214.
4. Kassay, L.; Slaughter, D.C.; Molnar, S. *Hungarian Robotic Apple Harvester*; ASAE: St. Joseph, MI, USA, 1992.
5. Bulanon, D.M.; Burks, T.F.; Alchanatis, V. *Study on Fruit Visibility for Robotic Harvesting*; ASABE Paper No. 07-3124; ASABE: St. Joseph, MI, USA, 2007.
6. Tabb, A.; Peterson, D.; Park, J. *Segmentation of Apple Fruit from Video via Background Modeling*; ASAE: St. Joseph, MI, USA, 2006.
7. Bargoti, S.; Underwood, J.P. Image segmentation for fruit detection and yield estimation in apple orchards. *J. Field Robot.* **2017**, *34*, 1039–1060. [[CrossRef](#)]
8. Tanigaki, K.; Fujiura, T.; Akase, A.; Imagawa, I. Cherry Harvesting Robot. In Proceedings of the 3rd IFAC International Workshop on Bio-robotics, Information Technology and Intelligent Control for Bioproduction Systems, Sapporo, Japan, 9–10 September 2006; pp. 254–260.
9. Van Henten, E.J.; Hemming, J.; Van Tuyl, B.A.J.; Kornet, J.G.; Bontsema, J. Collision-free Motion Planning for a Cucumber Picking Robot. *Biosyst. Eng.* **2003**, *86*, 135–144. [[CrossRef](#)]
10. Payne, A.B.; Walsh, K.B.; Subedi, P.P.; Jarvis, D. Estimation of mango crop yield using image analysis—segmentation method. *Comput. Electron. Agric.* **2013**, *91*, 57–64. [[CrossRef](#)]

11. Qureshi, W.S.; Payne, A.; Walsh, K.B.; Linker, R.; Cohen, O.; Dailey, M.N. Machine vision for counting fruit on mango tree canopies. *Precis. Agric.* **2017**, *18*, 224–244. [[CrossRef](#)]
12. Harrell, R.C.; Adsit, P.D.; Pool, T.A.; Hoffman, R. *The Florida Robotic Grove Lab*; ASAE Paper no. 88-1578; ASAE: St. Joseph, MI, USA, 1988.
13. Hannan, M.W.; Burks, T.F. *Current Developments in Automated Citrus Harvesting*; ASAE Paper no., 043087; ASAE: St. Joseph, MI, USA, 2004.
14. Chaivivatrakul, S.; Moonrinta, J.; Dailey, M.N. Towards Automated Crop Yield Estimation-Detection and 3D Reconstruction of Pineapples in Video Sequences. In Proceedings of the Fifth International Conference on Computer Vision Theory and Applications, Angers, France, 17–21 May 2010; pp. 180–183.
15. Chaivivatrakul, S.; Dailey, M.N. Texture-based fruit detection. *Precis. Agric.* **2014**, *15*, 662–683. [[CrossRef](#)]
16. Kondo, N.; Nishitsuji, Y.; Ling, P.P.; Ting, K.C. Visual Feedback Guided Robotic Cherry Tomato Harvesting. *Trans. ASAE* **1996**, *39*, 2331–2338. [[CrossRef](#)]
17. Clement, J.; Novas, N.; Gazquez, J.A.; Manzano-Agugliaro, F. An active contour computer algorithm for the classification of cucumbers. *Comput. Electron. Agric.* **2013**, *92*, 75–81. [[CrossRef](#)]
18. Novas, N.; Álvarez-Bermejo, J.A.; Valenzuela, J.L.; Gázquez, J.A.; Manzano-Agugliaro, F. Development of a smartphone application for assessment of chilling injuries in zucchini. *Biosyst. Eng.* **2019**, *181*, 114–127. [[CrossRef](#)]
19. Whittaker, D.; Miles, G.E.; Mitchell, O.R.; Gaultney, L.D. Fruit location in a partially occluded image. *Trans. ASAE* **1987**, *30*, 591–596. [[CrossRef](#)]
20. Pla, F. Recognition of partial circular shapes from segmented contours. *Comput. Vis. Image Undst.* **1996**, *63*, 334–343. [[CrossRef](#)]
21. Grasso, G.M.; Recce, M. Scene analysis for an orange harvesting robot. *AI Appl.* **1997**, *11*, 9–15.
22. Qureshi, W.S.; Satoh, S.I.; Dailey, M.N.; Ekpanyapong, M. Dense segmentation of textured fruits in video sequences. In Proceedings of the IEEE International Conference on Computer Vision Theory and Applications (VISAPP), Lisbon, Portugal, 5–8 January 2014; Volume 2.
23. Jiménez, A.R.; Ceres, R.; Pons, J.L. A survey of computer vision methods for locating fruit on trees. *Trans. ASAE* **2000**, *43*, 1911. [[CrossRef](#)]
24. Sharifi, M.; Rafiee, S.; Keyhani, A.; Jafari, A.; Mobli, H.; Rajabipour, A.; Akram, A. Some physical properties of orange (var. Tompson). *Int. Agrophys.* **2007**, *21*, 391–397.
25. Hannan, M.W.; Burks, T.F.; Bulanon, D.M.A. A machine vision algorithm combining adaptive segmentation and shape analysis for orange fruit detection. *Agric. Eng. Int. CIGR J.* **2010**, *11*, 1281.
26. Zapata-Sierra, A.J.; Manzano-Agugliaro, F. Controlled deficit irrigation for orange trees in Mediterranean countries. *J. Clean. Prod.* **2017**, *162*, 130–140. [[CrossRef](#)]
27. Plebe, A.; Grasso, G. Localization of spherical fruits for robotic harvesting. *Mach. Vis. Appl.* **2001**, *13*, 70–79. [[CrossRef](#)]
28. Gongal, A.; Amatya, S.; Karkee, M.; Zhang, Q.; Lewis, K. Sensors and systems for fruit detection and localization: A review. *Comput. Electron. Agric.* **2015**, *116*, 8–19. [[CrossRef](#)]
29. Lefsky, M.A.; Harding, D.; Cohen, W.B.; Parker, G.; Shugart, H.H. Surface lidar remote sensing of basal area and biomass in deciduous forests of eastern Maryland, USA. *Remote Sens. Environ.* **1999**, *67*, 83–98. [[CrossRef](#)]
30. Huang, H.; Li, Z.; Gong, P.; Cheng, X.; Clinton, N.; Cao, C.; Wenjian, N.; Wang, L. Automated methods for measuring DBH and tree heights with a commercial scanning lidar. *Photogramm. Eng. Remote Sens.* **2011**, *77*, 219–227. [[CrossRef](#)]
31. Sola-Guirado, R.R.; Bayano-Tejero, S.; Rodríguez-Lizana, A.; Gil-Ribes, J.A.; Miranda-Fuentes, A. Assessment of the Accuracy of a Multi-Beam LED Scanner Sensor for Measuring Olive Canopies. *Sensors* **2018**, *18*, 4406. [[CrossRef](#)] [[PubMed](#)]
32. Miranda-Fuentes, A.; Rodríguez-Lizana, A.; Gil, E.; Agüera-Vega, J.; Gil-Ribes, J.A. Influence of liquid-volume and airflow rates on spray application quality and homogeneity in super-intensive olive tree canopies. *Sci. Total Environ.* **2015**, *537*, 250–259. [[CrossRef](#)] [[PubMed](#)]
33. Miranda-Fuentes, A.; Llorens, J.; Rodríguez-Lizana, A.; Cuenca, A.; Gil, E.; Blanco-Roldán, G.L.; Gil-Ribes, J.A. Assessing the optimal liquid volume to be sprayed on isolated olive trees according to their canopy volume. *Sci. Total Environ.* **2016**, *568*, 296–305. [[CrossRef](#)]

34. Miranda-Fuentes, A.; Rodríguez-Lizana, A.; Cuenca, A.; González-Sánchez, E.J.; Blanco-Roldán, G.L.; Gil-Ribes, J.A. Improving plant protection product applications in traditional and intensive olive orchards through the development of new prototype air-assisted sprayers. *Crop Prot.* **2017**, *94*, 44–58. [[CrossRef](#)]
35. Verroust, A.; Lazarus, F. Extracting skeletal curves from 3D scattered data. *Vis. Comput.* **2000**, *16*, 15–25. [[CrossRef](#)]
36. Mizoue, N.; Masutani, T. Image analysis measure of crown condition, foliage biomass and stem growth relationships of Chamaecyparis obtusa. *For. Ecol. Manag.* **2003**, *172*, 79–88. [[CrossRef](#)]
37. Pfeifer, N.; Gorte, B.; Winterhalder, D. Automatic reconstruction of single trees from terrestrial laser scanner data. In Proceedings of the 20th ISPRS Congress: Geo-imagery Bridging Continents, Istanbul, Turkey, 12–23 July 2004; pp. 114–119.
38. Phattaralerphong, J.; Sinoquet, H. Tree Analyser: Software to Compute Tree Structure Parameters from Photographs. User Manual. PIAF-INRA. 2007. Available online: <http://www2.clermont.inra.fr/piaf/eng/download/download.php> (accessed on 1 August 2019).
39. Phattaralerphong, J.; Sinoquet, H. A method for 3D reconstruction of tree crown volume from photographs: Assessment with 3D-digitized plants. *Tree Physiol.* **2005**, *25*, 1229–1242. [[CrossRef](#)]
40. Bailey, B.N.; Ochoa, M.H. Semi-direct tree reconstruction using terrestrial LiDAR point cloud data. *Remote Sens. Environ.* **2018**, *208*, 133–144. [[CrossRef](#)]
41. Lim, M.; Rosser, N.; Allison, R.; Petley, D. Erosional processes in the hard rock coastal cliffs at Staithes. *Geomorphology* **2010**, *114*, 12–21. [[CrossRef](#)]
42. García-Gómez, I.; de Gorostiza, M.; Moraza, A. Láser escáner y nubes de puntos. Un horizonte aplicado al análisis arqueológico de edificios. *Arqueol. Arquit.* **2010**, *30*, 25–44. [[CrossRef](#)]
43. Benedí, J.; Moreno, J.; Molina, C.; Arguedas, G.; Serrano, S.; Oliván, A. Utilización de técnicas de láser escáner terrestre en la monitorización de procesos geomorfológicos dinámicos: El manto de nieve y heleros en áreas de montaña. *Cuadernos de investigación geográfica* **2013**, *39*, 335–357.
44. Ball, G.H.; Hall, D.J. *ISODATA, A Novel Method of Data Analysis and Pattern Classification*; Stanford Research Inst.: Menlo Park, CA, USA, 1965.
45. Ridler, T.W.; Calvard, S. Picture thresholding using an iterative selection method. *IEEE Trans. Syst. Man Cybern.* **1978**, *8*, 630–632.
46. Unay, D.; Gosselin, B. Thresholding-based segmentation and apple grading by machine vision. In Proceedings of the 13th European Signal Processing Conference, Antalya, Turkey, 4–8 September 2005.
47. Shruthi, G. Automatic Detection and Classification of Apple-A Survey. In Proceedings of the Third International Conference on Current Trends in Engineering Science and Technology (ICCTEST 2017), Bangalore, India, 5–7 January 2017; Don Bosco Institute of Technology: Dbit, Bangalore, India.
48. Lloyd, S.P. Least squares quantization in PCM. *IEEE Trans. Inform. Theory* **1982**, *28*, 129–137. [[CrossRef](#)]
49. Cormen, T.H.; Leiserson, C.E.; Rivest, R.L.; Stein, C. *Introduction to Algorithms*, 3rd ed.; The MIT Press: London, UK, 2009; pp. 248–300.
50. Goldberg, K.M.; Iglewicz, B. Bivariate extensions of the boxplot. *Technometrics* **1992**, *34*, 307–320. [[CrossRef](#)]
51. R Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria, 2015.
52. Stein, M.; Bargoti, S.; Underwood, J. Image Based Mango Fruit Detection, Localisation and Yield Estimation Using Multiple View Geometry. *Sensors* **2016**, *16*, 1915. [[CrossRef](#)] [[PubMed](#)]
53. Liu, J.; Yuan, Y.; Zhou, Y.; Naz Syed, T. Experiments and Analysis of Close-Shot Identification of On-Branch Citrus Fruit with RealSense. *Sensors* **2018**, *18*, 1510. [[CrossRef](#)] [[PubMed](#)]
54. Tabatabaeefar, A.; Vefagh-Nematolahee, A.; Rajabipour, A. Modeling of orange mass based on dimensions. *Agric. Sci. Technol.* **2000**, *2*, 299–305.

