TITLE: OLIVE FRUIT IDENTIFICATION IN OLIVE-TREE IMAGES, TAKEN IN INTENSIVE OLIVE ORCHARDS, BY MEANS OF MORPHOLOGY-BASED REGION PROPOSAL AND CONVOLUTIONAL NEURAL NETWORKS

AUTHORS: ARTURO AQUINO, JUAN MANUEL PONCE, JOSÉ MANUEL ANDÚJAR

AFFILIATION: DEPARTAMENTO DE INGENIERÍA ELECTRÓNICA, SISTEMAS INFORMÁTICOS Y AUTOMÁTICA. UNIVERSIDAD DE HUELVA. CARRETERA HUELVA-PALOS, S/N, 21810 PALOS DE LA FRONTERA (ESPAÑA).

CONTACT: {ARTURO.AQUINO, JMPONCE.REAL, ANDUJAR}@DIESIA.UHU.ES

CORRESPONDING AUTHOR: ARTURO AQUINO.

HIGHLIGHTS:

- Accurate yield estimation increases earnings and stabilises market price.
- Traditional estimations are becoming inaccurate as climate variability increases.
- First image dataset (OLIVEnet) to develop solutions for olive-fruit identification.
- First algorithm able to identify olive fruits in tree images taken in the field.
- Accuracy of results supports viability of yield estimation using image analysis.

ABSTRACT:

Accurate yield estimation is a greatly desired objective in oliviculture due to the high economic value of its production. This paper presents a methodology aimed at achieving that end. It comprises an artificial-vision algorithm able to detect visible olives in digital images of olive trees captured directly in the field, at night-time and with artificial illumination; these images were taken in an intensive olive orchard of the Picual *Olea europaea* L. variety in September 2018 (two months prior to harvesting). Regarding the methodology, first, the images are pre-processed to generate a set of sub-images with high probability of containing an olive, thus reducing the search space in a magnitude of 10^3. Next, these sub-images are classified by a

convolutional neural network (CNN) as *olive*, if they are centred in an olive fruit, or as *other* in any other case (even if they contain peripheral fruits). To train and validate the CNN, a special database called OLIVEnet was compiled with two disjoint sets integrating these sub-images. A training and a validation set was built with 234,168 and 299,946 *olive* and *other* sub-images, respectively. Five different CNN topologies were tried, correctly classifying the best performing one 83.13% of *olive* instances, with 84.80% of precision, and 99.12% (*SP*) of *other* instances; measured accuracy and F_1 Score were 0.9822 and 0.8396, respectively. As far as the authors' knowledge goes, this article presents the first image analysis approach to automatically identify olive fruits directly on the whole tree image. The obtained results constitute a first and solid step towards the implementation of an automatic system for yield estimation of olive orchards.

KEYWORDS: OLIVENET, CONVOLUTIONAL NEURAL NETWORKS, YIELD ESTIMATION, OLIVE, PRECISION AGRICULTURE.

1. Introduction

Olive cultivation (*Olea europaea* L.), and its associated market, is an outstanding economic engine for most of the Mediterranean basin, as this area provided 64.16% of the total amount of olives produced worldwide (FAOSTAT 2018).

Olive-fruit yield estimation is a valuable supporting tool for the sector (Orlandi et al. 2010), as it is shown to favour the practical improvement of aspects such as: olive oil transformation efficiency, stock management, or human resources management optimization for harvesting (Aguilera and Ruiz-Valenzuela 2014). Traditionally, yield estimation has been addressed by farmers, by observing the amount of visible fruit directly in the field. Notwithstanding, the huge extension of olive orchards, which prevents the exhaustive analysis, along with the subjective nature of this evaluation, impoverish the impact of these projections. This is remarkably visible when analysing olive oil's price instability, which is consequence of inaccuracy of yield expectations (European Commission 2011).

As an alternative to this classical approach, estimation models built with indirect information, such as meteorological or pollination variables, have been explored (Fornaciari et al. 2005; Galán et al. 2004; Minero et al. 1998; Galán et al. 2008; Oteros et al. 2014; Fabio et al. 2010), and are even in practical use for years as. An example is the EU's MARS Crop Yield Forecasting System (MCYFS)[1], which is being exploited since 1993 to predict production of a great set of crops, including olive orchards. However, the main weakness of this approach emerges from the intrinsic stochastic nature of the variables involved, which compromises model reproducibility and accuracy (Van der Velde and Nisini 2019).

Computer vision is being experimented as an alternative to implement yield estimation systems. This new approach aims to mimic the ancient procedure carried out by farmers of estimating yield by visually analysing the amount of visible fruit. In this sense, this paper is focused on the

[1] MCYFS Wiki: https://marswiki.jrc.ec.europa.eu/agri4castwiki/index.php/Welcome_to_WikiMCYFS

investigation, development and testing of an image analysis methodology able to individually identify and count the visible olive fruits present in images of olive trees taken in the field.

Analog initiatives can be found in the literature for a variety of crops. Indeed, computer vision proposals for in-the-field fruit recognition can be found in the literature for the case of vineyards (Font et al. 2015; Nuske et al. 2014; Liu et al. 2013; Aquino et al. 2018; Millan et al. 2018), and orchard crops such as apples (Bargoti and Underwood 2017a; Nguyen et al. 2016; Bargoti and Underwood 2017b), mangoes (Bargoti and Underwood 2017b; Qureshi et al. 2017), sweet-peppers (Vitzrabin and Edan 2016; Bac et al. 2013), almonds (Bargoti and Underwood 2017b; Qureshi et al. 2017; Vitzrabin and Edan 2016; Bac et al. 2013; Hung et al. 2013) or tomatoes (Yamamoto et al. 2014; Zhao et al. 2016), among other. In general, the solution built for recognising a specific type of fruit is hardly applicable to others, due to the visual differences resulting from the specific features of every crop and fruit type. Regarding the detection of olives in olive-tree images it is especially remarkable, as there are many distinctive features involved making this case rather unique and challenging: 1) the number of visible olive fruits per tree can be counted in hundreds or even in thousands; 2) olive-fruit colour is similar to that of leaves; 3) olive-fruit visual occlusions with other fruits, leaves or branches is very frequent; 4) olive-fruit size is tiny with respect to that of olive trees. Several developments for olive-fruit segmentation can be found in the literature. However, most of them are aimed at improving post-harvest olive-fruit classification, so they analyse images of collected olive fruits placed over homogeneous backgrounds. Examples of these works are those by Ponce et al. (Ponce et al. 2019; Ponce et al. 2018), focused in estimating olive-fruit mass and size, and those by Diaz et al. (2004) and Puerto et al. (2015), who dealt with olive-fruit classification according to their surface condition. An exception is the development by Gatica et al. (2013), who presented an image analysis algorithm based on neural networks to detect olive fruits in tree brunches. Notwithstanding, its practical application is rather limited, as brunches were cut prior to be photographed over a white homogeneous background.

To the best of the authors' knowledge, this paper presents the first approach to automatically identify individual olive fruits in images of the whole olive tree, directly taken in the field at night-time, using artificial illumination. First, for every image, a reduced set of sub-images representing olive-fruit candidates is obtained by characterising, with mathematical morphology techniques, the pattern of light reflection occurred on the fruits' surface; this scheme allows to reduce the size of the potential search space in a magnitude of 10^3. Then, the sub-images are pre-processed to optimize the performance of a convolutional neural network in the task of deciding the presence or absence of a centred olive fruit.

The rest of the paper is structured as follows: section 2 characterises the field experimental design and image acquisition; section 3 presents the image analysis algorithm, by describing the designed image pre-processing and all related to the subsequent classification using convolutional neural networks; results are detailed and discussed in section 4; the paper ends stating the main conclusions derived from the investigation and pointing out the future work.

2. Field experimental design and image acquisition

A set of 36 tree images of the Picual *Olea europaea* L. variety was acquired in September 2018 (two months prior to harvesting) in an intensive commercial olive orchard placed in Gibraleón (37°20'09.2"N 7°02'19.8"W), province of Huelva (southwest of Andalusia, Spain); row and olive-tree spacing was 7 m and 5.5 m, respectively. The photographed individuals were selected to cover the maximum available variability in terms of olive-fruit productive capacity.

Images were taken at night-time with the artificial illumination provided by a halogen spotlight of 500 W. This decision is supported by the following three reasons: 1) a future robotic system would not interfere with daily agronomic activities during scouting missions; 2) the whole robotic system would operate under more favourable temperature conditions, which will result in better reliability; 3) the conclusions of previous related works faced by the authors of this paper and other, such as those by Aquino et al. (2018) and Nuske et al. (2014), stand for an optimum control of illumination in image capturing, as it produces images with more favourable

conditions for their analysis (illumination homogenising, shadow minimising, elimination of undesired planes from the image, etc.).

Image acquisition was performed manually using the Sony α7II mirror-less RGB camera (Sony Corp., Tokyo, Japan), equipped with a Zeiss 24/70 mm (Carl Zeiss AG, Oberkochen, Germany), and mounted on a tripod. The camera was set in manual mode, configuring the aperture in f/14, shutter speed in 1/200 s, ISO sensitivity in 800 and focus in manual mode. The images were saved in JPEG format with minimum compression, at a resolution of 6,000 × 3,376 and 24 bits of colour depth (8 bits per RGB channel). The distance at which images were taken varied between 2 m and 4 m, depending on the size of the olive tree, with the aim of properly covering the whole canopy in every case. As an example, Fig. 1 shows an image captured under the described conditions.

Fig. 1.- Olive tree image from the Picual variety, object of the present study, and taken following the precepts described in section 2.

3. Image analysis methodology for olive-fruit identification

The methodology comprises the use of a Convolutional Neural Network (CNN) to individually identify olive fruits visible in olive-tree images such as the one shown in Fig. 1. To this end, a specialised pre-processing was designed to effectively: 1) improve starting image conditions; 2) reduce the potential search space by finding a set of sub-images with high probability of containing a centred olive-fruit; and 3) build an optimum configuration for the sub-images to

favour the performance of a CNN when classifying them as containing a centred olive or not. Fig. 2 shows a conceptual diagram illustrating the main steps comprising the designed methodology. The methodology described throughout this paper was implemented using the Matlab R2018b platform and its Image Processing and Deep Learning toolboxes, release 2018a (The Math-Works Inc., Natick, Massachusetts, USA).

3.1 Image pre-processing

Let I be a digital image of an olive tree captured following the principles described in section 2. This initial image is transformed to the CIE 1976 $L^*a^*b^*$ colour space (Connolly and Fliess 1997), as it allows to independently analyse the illumination (L^* channel) and colour (a^* and b^* channels) components. Indeed, these channels configure a spherical space where L^* is the vertical component determining illumination, and a^* and b^*, for any given value of L^*, represent a cartesian coordinate system in whose origin a chromatic circle is defined.

A luminance 8-bit image L is computed by linearly transforming L^* (defined in $[0, 100]$) to take values in $[0, 255]$:

$$L = \frac{L^*}{100} \times 255 \tag{1}$$

On the other hand, a^* and b^* configure a chromatic circle in such a way that, using polar coordinates instead of cartesian ones, a^* and b^* select a colour by specifying a distance and an angle, meaning the former its saturation and the latter its hue. Being hue valuable information for the fundamental aim of this work, which is to identify olive-fruits present in images such as that shown in Fig. 1, channels a^* and b^* are transformed to produce a unified image H containing this information:

$$\theta = arctangent\left(\frac{b^*}{a^*}\right) \tag{2}$$

where θ is a hue channel, as it contains the angles resulting from transforming the cartesian coordinates of the system formalised by a^* and b^*. To finally obtain the 8-bit hue image *H*, the angles are normalised and linearly transformed to take values from 0 to 255:

$$H = \left(\frac{\theta}{2 \times \pi}\right) \times 255 \quad (3)$$

There exist colour spaces providing independent hue channels, such as HSV or HSI (Sonka et al. 2014). Notwithstanding, after testing, they were found to give hue channels with noticeably lower contrast than the one obtained by the described procedure.

'Salt and pepper' noise is reduced by applying a circular average filtering, implemented by a 11×11 convolutional kernel k, to images L and H:

$$L' = L * k; H' = H * k \quad (4)$$

where the circle of filter activation has 11 pixels in diameter. The reduced size of this filter compared to that of the image, 11×11 vs $6{,}000 \times 3{,}376$, provides certain tolerance in setting its value to achieve the desired effect. Moreover, circular filtering favours removing analogous patterns present in the olive fruits.

Next, a set of seeds is calculated from L'. These seeds will serve as reference to subsequently generate a set of sub-images (or candidates), thus reducing the search space. To this end, every seed will be an aggregation of neighbour pixels formally known as connected component (CC), which will represent the occurrence of a local regional maximum of illumination. Thanks to the

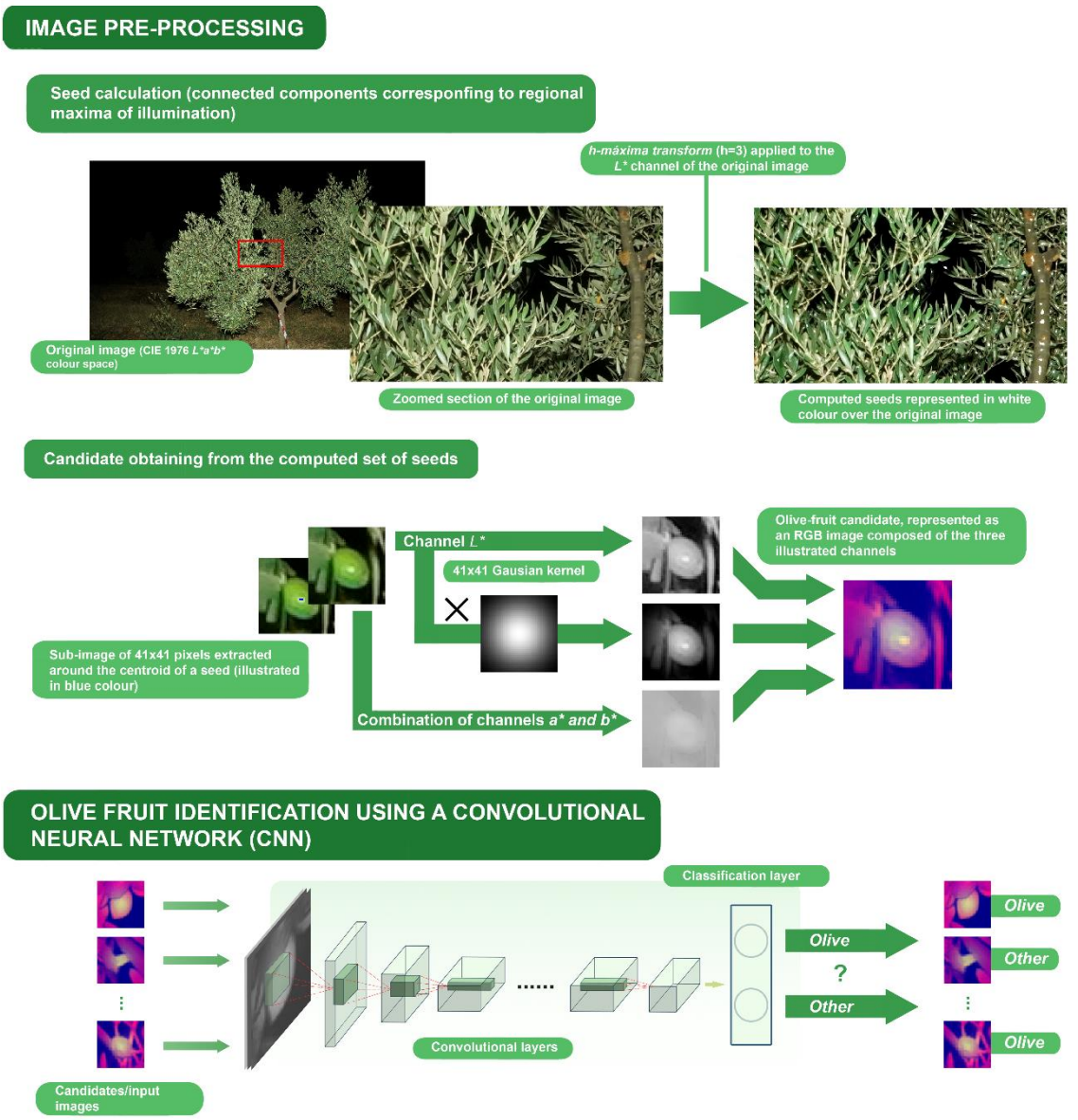

Fig. 2.- Conceptual diagram scheme of the methodology for olive-fruit identification in olive-tree images.

Lambert's cosine law (Smith 2000), a convex surface, such as that of olive fruits, is well known to generate an intense circular pattern of light reflection. On this basis, by finding significant regional local maxima on L' (which directly comes from the illumination channel L^*), favours locating olive fruits present in the image. With this aim, a three-step procedure is designed on the basis of Mathematical Morphology's principles. This theory provides powerful operators to explore local properties built from the concept of pixel connectivity, and the response of neighbouring pixels when they are proven to a filter with a known shape called structuring

element. The first step within the procedure comprises supressing from L' irrelevant maxima regions, which are those not high enough with respect to their surroundings to be considered fundamental light reflection zones. The *h*-maxima morphological transform allows to achieve this end by supressing those regional maxima with an elevation less than or equal to scalar h. Mathematically, it is formalised by the morphological reconstruction R of image L' from marker $L' - h$ (consult details about the reconstruction operator in Soille (2004)),

$$L_{filt} = R_{L'}(L' - h); h = 3 \quad (5)$$

Then, surviving regional maxima are considered significant and, therefore, extracted from the computed image. It is achieved by subtracting from L_{filt} the result of supressing from it all the remaining regional maxima:

$$L_{RM} = L_{filt} - R_{L_{filt}}(L_{filt} - 1) \quad (6)$$

Note that the right term of the subtraction is the *h*-maxima transform, setting *h* to value 1. This value allows to remove all remaining regional maxima from L_{filt}, as all of them will fulfil to have, at least, that height. Finally, the extracted regions are segmented by binarizing L_{RM}:

$$L_{RMbin}(x, y) = \begin{cases} 255 \; if \; L_{RM}(x, y) > 0 \\ 0 \qquad in \; other \; case \end{cases} \quad (7)$$

In analogous situations of application investigated by authors of this paper, the optimum value for parameter h was discussed (Aquino et al. 2018; Aquino et al. 2017). These works concluded 3 or 4 to be optimum values, although not exclusive, as satisfactory results were also found with slight variations; for the experimentation described in this paper, a value of 3 was set. Finally, the set of regional maxima of illumination (or seeds) is composed of the connected components, CC_i, existing in the binary image L_{RMbin}:

$$S_{RM} = \{CC_i \subseteq L_{RMbin}\} \quad (8)$$

Fig. 3 illustrates the results derived from calculating the set of connected components of illumination, by applying the described methodology.

The next operation consists in building a set of sub-images from L' and H', respectively, centred in the found seeds; these sets will be referred as S_L and S_H, respectively. To this effect, for every connected component $CC_i \in S_{RM}$, a 41×41 sub-image s_L^i and s_H^i from L' and H' is extracted around the centroid of CC_i, thus fulfilling:

$$\#(S_L) = \#(S_H) = \#(S_{RM})$$

$$S_L = \{s_L^i | s_L^j \subset L', (l,m) = ctr(CC_j) \Rightarrow L'(l,m) = s_L^j(21,21)\} \quad (9)$$

$$S_H = \{s_H^i | s_H^j \subset H', (l,m) = ctr(CC_j) \Rightarrow H'(l,m) = s_H^j(21,21)\}$$

where $ctr(CC_j)$ denotes the coordinates of the centroid of the *j*-th connected component.

Olive trees were photographed at a distance varying between 2 m and 4 m to capture the whole canopy independently from their size. Furthermore, olive trees cultivated under a traditional or an intensive scheme, which is the case of this study, develop a considerable canopy volume, which makes their first plane to be noticeably closer to the camera than their back plane. Consequently, olive- fruit appearance in terms of size may vary within a given image and among images. Taking this into account, the size of the sub-images was stablished to contain an olive fruit in any case. On the other hand, considering that the centre of an olive-fruit can be located anywhere within the image, the potential search space for the image resolution managed in this work is composed of 20,256,000 candidates. As, in average, 14,499 sub-images were produced from pre-processing an image, the search space was reduced in a magnitude of 10^3.

Once sub-images from the sets S_L and S_H are obtained, the last step of the pre-processing consists in finding such a combination of them favouring the performance of a CNN in the task of deciding if a sub-image contains a fruit or not. Furthermore, the net has also to be capable of identifying if the sub-image under evaluation is centred in an olive fruit to validate it, or in any other element, such as a leaf, to discard it. In this sense, for the case of sub-images centred in non-fruit elements, it can't be guaranteed that olives never appear, totally or partially, in peripheral parts (Fig. 4). In these cases, the net has also to be flexible enough to understand that

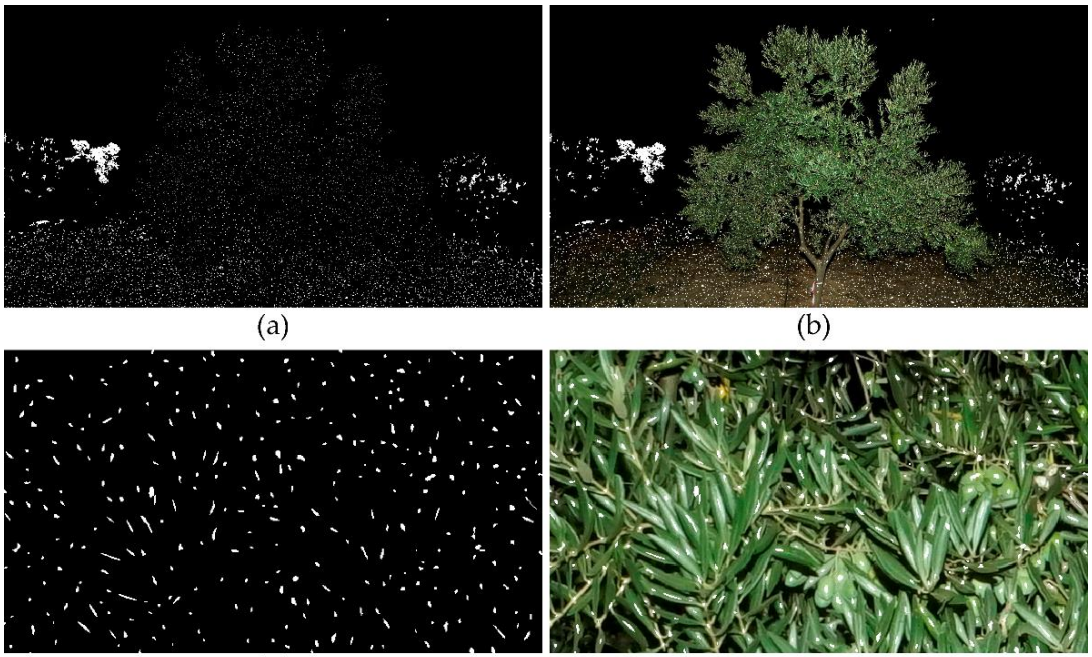

Fig. 3.- Illustration of the procedure to compute the connected components of illumination, or seeds, used to reduce the search space: (a) seeds represented in white over a black background and (b) over the original image; (c) and (d) are zoomed sections of the same region of (a) and (b), respectively.

the sub-image must be classified as not containing an olive. To favour this performance, a new sub-set called S_{LG} is created from S_L, by multiplying its sub-images s_L^i, element by element, by a normalised Gaussian matrix G of size 41×41 and $\sigma = 9.5$:

$$S_{LG} = \left\{ s_{LG}^i \middle| s_{LG}^j(k,l) = s_L^j(k,l) \times G(k,l); \forall k \forall l, 1 \leq k,l \leq 41;\ 1 \leq j \leq \#(S_{RM}) \right\} \quad (10)$$

Indeed, sub-image s_{LG}^i contains the same information as s_L^i, but weighted in decreasing relevance from the centre to the border (see Fig. 4).

Finally, the union of the three ordered sets, S_L, S_H and S_{LG}, configures the search space for image *I* from which they derive. Thus, each group of three *i*-th elements can be considered as a three-dimensional sub-image of 41×41, where s_L^i provides illumination information, s_{LG}^i weights the relevance of this information according to its position within the image, and s_H^i gives knowledge related to colour (see Fig. 5).

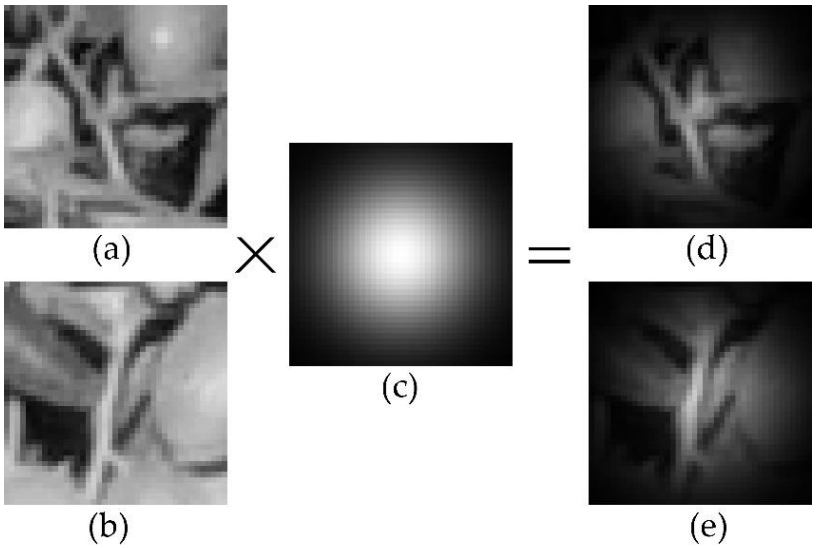

Fig. 4.- (a) and (b) Sub-images from S_L containing olives, but centred in a regional maximum of illumination produced by a branch in both cases; (c) representation of a Gaussian matrix of size 41×41 and $\sigma = 9.5$; (d) and (e) sub-images from S_{LG} obtained by multiplying, element by element, (a) and (b) by (c), respectively. Note how the relevance of peripherical olives is reduced in both images.

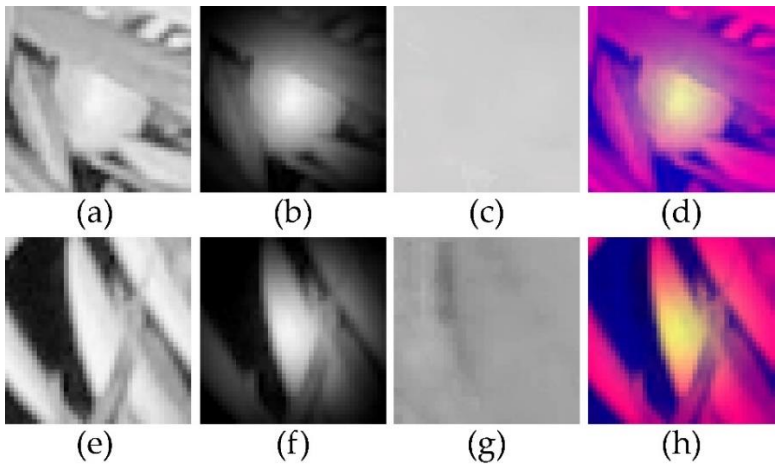

Fig. 5.- (a) - (d) sub-images s_L^i, s_{LG}^i, s_H^i and the RGB representation of their combination, respectively; they result from the *i*-th seed given by a regional maximum of illumination produced by an olive; (e) - (h) sub-images s_L^j, s_{LG}^j, s_H^j and their RGB representation, respectively; they derive from the *j*-th seed given by a regional maximum of illumination produced by a leaf.

3.2 Methodology for olive fruit identification

This stage of the methodology deals with the configuration, training and validation of a CNN to accurately classify the sub-images generated by the pre-processing as olive (positive case) or other (negative case).

3.2.1 CNN architecture

CNNs have experimented a tremendous development, especially during the past decade, as consequence of various factors, such as the huge increase in terms of computational capacity achieved by affordable GPUs, or the popularisation of datasets with massive quantities of labelled images, publicly available for all the scientific community (i.e., ImageNet (Deng et al. 2009) or CIFAR-10 (Krizhevsky 2009)). However, CNNs have emerged as a key booster due to their own philosophy. Indeed, whereas classical approaches using fully connected neural networks dealt with the difficulty of designing significant mathematical descriptors to train the net to solve a specific problem (Kumar and Bhatia 2014), CNNs have the ability of automatically learning these descriptors, so they can be directly fed with entire images.

In general, CNNs are composed by two main structures. The first one, which is in charge of feature extraction, is essentially conformed by a sequence of layers implementing convolutional filters (convolutional layers), alternated with normalisation layers, activation layers (typically rectified linear units) and layers for reducing the spatial dimensionality (pooling layers). The output of this structure is connected to the input of the second structure, which typically is a fully connected multilayer perceptron. Thus, this structure combines the features obtained by the previous one to learn and perform image classification.

Theoretically, there are infinite possible CNN configurations depending on the number of layers, or the combination, configuration and connection of them, among many other aspects. Moreover, no deterministic evidence can be a-priori inferred about what is the optimum CNN configuration for a given problem to be solved. Thus, performance of a CNN design is ultimately assessed empirically, by facing the net against a classification challenge. To this effect, global

competitions have become a benchmark, accepted by the scientific community, to compare under normalised conditions performance of the different CNN proposals. Probably, the reference of these competitions in terms of prestige is ImageNet Large Scale Visual Recognition Competition (ILSVRC), as most of their successful winners have become almost a standard, showing great ability to solve complex problems. This is the case of, AlexNet (Krizhevsky et al. 2012) (won ILSVRC in 2012), VGGNet (Simonyan and Zisserman 2014) (won in 2015), Inception (Szegedy et al. 2015) (won in 2014), ResNet (He et al. 2015) (shared the winner title with VGGNet in 2015) and InceptionResNet (Szegedy et al. 2017), which is the combination of the Inception and ResNet approaches.

AlexNet (Krizhevsky et al. 2012) is the older of these architectures and supposed a significant advance upon previous alternatives. From its presentation, the other resulted as alternatives to improve its performance under the general choice of being deeper structures, as it increases non-linearity and allows to get better feature representation. The VGGNet architecture (Simonyan and Zisserman 2014) was formulated to be denser, but in such a way that layer increasing with respect to AlexNet would not involve a proportional increase in the number of parameters. The Inception architecture (Szegedy et al. 2015) takes the name from the introduced Inception module. Under the basis that most of the activations in the net were irrelevant, this module allowed to effectively connect only the most important activations at the output to the input of the following module; the architecture also postulates to apply convolutions of different sizes in the same layer, thus pursuing flexibility in modelling patterns at varied scales. The ResNet paradigm (He et al. 2015) defines and exploits the concept of learning residue to improve and optimise convergence to the optimum solution; basically, a learning residue is the difference between that learnt at the input and at the output of a given layer. Finally, InceptionResNet (Szegedy et al. 2017) unifies the most relevant benefits of both characteristics. For a deeper study of CNNs, the reader is encouraged to consult references (Gu et al. 2018; Lecun et al. 2015; Liu et al. 2016, Voulodimos et al. 2018).

The overviewed architectures generally have different versions. Table 1 summarizes the main characteristics of the concrete CNNs tested in this research.

Table 1

Main characteristics of the CNNs considered in this research.

	CNN architecture				
	AlexNet	VGG19	InceptionV3	ResNet-50	Inception-ResNetV2
Depth (layers)	8	19	48	50	164
Parameters (in millions)	61.0 M	144.0 M	23.9 M	25.6 M	55.9 M
Input image size (in pixels)	227×227	224×224	299×299	224×224	299×299

3.2.2 OLIVEnet Dataset configuration for CNN training and validation

Sub-images generated by pre-processing 15 out of the 36 olive-tree images were used for CNN training; the resting 21 were kept for external validation. To generate two sets of sub-images for training, labelled as *olive* (positive case) or *other* (negative case), the found connected components of illumination were represented in white colour in the original image (as shown in Fig. 6-(a)). Then, those representing an olive were manually painted in blue colour by using an image edition software; as it can be checked in Fig. 6-(b), this assignment was made independently from the degree of partial occlusion that olives could show with other artefacts. Consequently, for the 15 training images, the sub-images extracted around the centroid of a blue component were labelled as *olive*, while those extracted from white components were specified as *other*.

Fig. 6.- Procedure to label instances as *olive* or *other*, aimed at creating the annotated training and validation sets: (a) the connected components of illumination are represented in white colour in the original image; (b) those components corresponding to olives, independently from the degree of object occlusion, are painted in blue colour and labelled as *olive*, whereas the remaining white components are labelled as *other*.

From the total 218,605 training sub-images labelled, only 9,757 corresponded to class *olive*. Hence, as a first step to equilibrate both classes, every *olive* sub-image was rotated eleven times, 30º at each step, producing by this way 117,084 *olive* instances in total. The second step was to filter from *other* those instances less relevant in terms of class separability. In this sense, sub-images coming from connected components at the image background, out of the tree, contained very low contrasted, diffused and darkened objects, easily distinguishable from olives. Contrary, sub-images taken from inside the tree might even contain non-centred olives (situation illustrated in Fig. 4). With these precepts, case filtering was achieved by applying a k-nearest neighbours- based (Knn) clustering (Duda et al. 2012) to the set of standard deviation values, *SD*, calculated from the *L* channel of sub-images from class *other*:

$$SD_{other} = \left\{ sd^{i}_{other} \middle| sd^{j}_{other} = \sigma\left(s^{j}_{L}\right), s^{j}_{L} \epsilon S_{L}, s^{j}_{L} \; was \; labelled \; as \; other \right\} \quad (11)$$

Note that, the higher sd^{i}_{other} value is, the more textured its corresponding sub-image s^{i}_{L} is (see Fig. 7). Thus, Knn clustering was set to yield three clusters of significance, *high*, *medium* and *low*, giving the results detailed in Table 2. Finally, class *other* was configured with 29,999 instances from cluster *high* (100%), and 60,958 (~72%) and 26,125 (~28%) instances randomly selected from clusters *medium* and *low*, respectively.

Regarding external validation, the 21 olive-tree images kept to this effect generated 299,946 sub-images derived from connected components of illumination. These instances were labelled as *olive* or *other* to evaluate the performance of the CNN using the same procedure, illustrated in Fig. 6, employed to label the training ones. The process yielded 16,699 and 283,247 *olive* and *other* sub-images, respectively. Table 3 summarizes the configuration of OLIVEnet.

To encourage and favour investigation in the subject, OLIVEnet is publicly available to the scientific community, upon password request to the corresponding author of this work[2].

Table 2

Results of Knn clustering of the set SD_{other} of standard deviation values calculated from the *L* channel of sub-images from class *other*.

	Clustering result	
Cluster	Interval	Number of instances
high	$61.16 \leq sd^{i}_{other}$	29,999
medium	$14.79 < sd^{i}_{other} < 61.16$	84,791
Low	$sd^{i}_{other} \leq 14.79$	94,058

Table 3

OLIVEnet dataset organization for CNN training and validation.

	Olive-tree images	Sub-images generated by pre-processing		Sub-images after class balancing	
		Class	Figure	Class	Figure
Training	15	*olive*	9,757	*olive*	117,084[a]
		other	208,848	*other*	117,084[b]
		overall	218,605	overall	234,168

[2] OLIVEnet link: http://gofile.me/3YQig/S6SenXN7m

Validation	21	*olive*	16,699	-
		other	283,247	
		overall	299,946	

[a] Data augmentation by image rotation.

[b] Data filtering by Knn-based clustering.

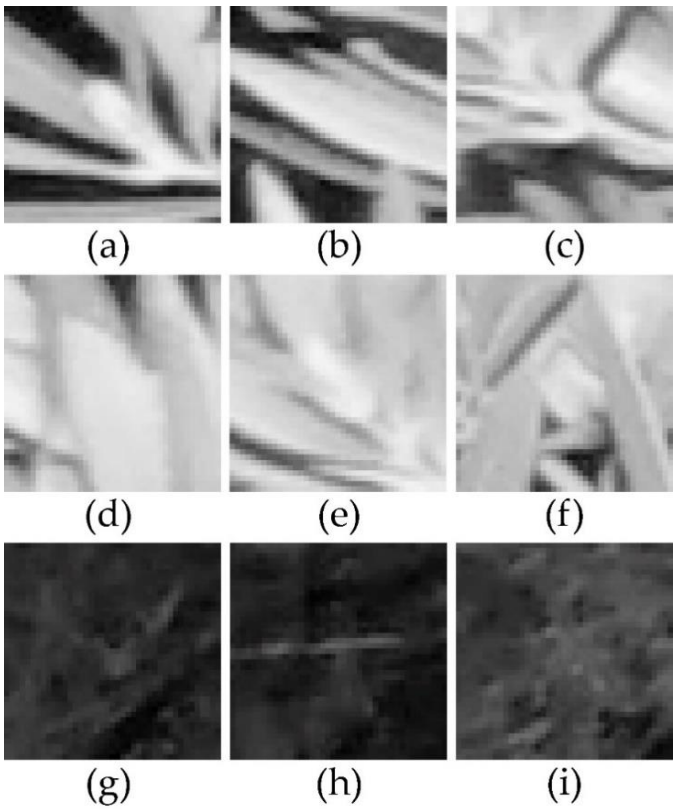

Fig. 7.- Illustration of Knn clustering results used to configure the training set: (a)-(c) images clustered as *high*, with standard deviation values of 69.70, 70.41 and 62.78, respectively; (d)-(f) images clustered as *medium*, having standard deviation values of 39.52, 32.61 and 36.56, respectively; (g)-(i) images from cluster *low*, with standard deviation values of 7.76, 7.74 and 8.91, respectively.

3.2.3 CNN training

The MATLAB framework was used to train the five considered CNNs for image classification. These classifiers were available pretrained with the thousands of images contained in the ImageNet dataset (Deng et al. 2009), and configured to provide predictions within the 1,000 annotated classes. To adapt the classifiers to the case of study of this research, i.e. to exclusively

classify instances as *olive* or *other*, but retaining the convolutional patterns a-priori 'known' by the implementations provided, the transfer learning strategy was followed (Shin et al. 2016). Basically, this approach consists in preserving the trained convolutional layers, while the fully connected one is substituted by a non-trained module adapted to the needed number of outputs (two in this case).

Table 4 details the main CNN training settings and milestones for the different architectures tried. The size of the training sub-images was adapted to match the input of each architecture. Convergence to an optimum result was judged for each of them by analysing the partial results offered by the loss function evolution during training; the learning rate was iteratively decreased to favour this convergence. Similarly, the mini-batch size used was not necessarily the same for the different CNNs either. Indeed, this parameter was adapted to each specific case for optimising the learning time. During training and before every epoch, data shuffle was performed to take advantage of the knowledge provided by the whole imagery.

Table 4

Summary of the main CNN training settings and milestones registered from the considered architectures.

	CNN architecture				
	AlexNet	VGG19	InceptionV3	ResNet-50	Inception-ResNetV2
Epochs	46	33	48	59	62
Iterations	144,514	221,036	160,739	276,815	519,634
Mini-batch size	75	35	70	50	28
Input image size (pixels)	227×227	224×224	299×299	224×224	299×299

Learning rate	10^{-3}-10^{-5}	10^{-3}-10^{-5}	10^{-3}-10^{-5}	10^{-3}-10^{-5}	10^{-3}-10^{-5}

3.2.4 Methodology for CNN performance evaluation

The 21 images building the external validation set were pre-processed, yielding 299,946 instances as specified in Table 3. The instances were analysed, and thus categorised as *olive* or *other*, with the five different CNNs proposed. Next, for every CNN variant, the given predictions were confronted to the manual annotations and true positives (TPs), true negatives (TNs), false positives (FPs), and false negatives (FNs) were calculated according to the following definitions:

- TP: *olive* instance classified as *olive* by the CNN.
- TN: *other* instance classified as *other* by the CNN.
- FP: *other* instance classified as *olive* by the CNN.
- FN: *olive* instance classified as *other* by the CNN.

This approach allowed to assess the performance of the different CNNs by using the following five metrics based on contingency tables for binary classification:

$$SE = \frac{TP}{TP + FN}; PR = \frac{TP}{TP + FP};\ SP = \frac{TN}{TN + FP}$$
$$ACC = \frac{TP + TN}{TP + FP + TN + FN}; F_1 = 2 \times \frac{PR \times SE}{PR + SE} \quad (12)$$

where, for a given CNN: 1) *SE* stands for Sensitivity, being this the rate of instances manually labelled as *olive* detected by the classifier; 2) *PR* denotes Precision, which calculates the hit rate achieved by the CNN when predicting the class *olive*; 3) *SP* is Specificity, which measures the rate of *other* instances predicted; 4) metric *ACC* means Accuracy, and provides a general hit rate when predicting the classes *olive* and *other*; and 5) F_1 represents F_1 Score, this giving the harmonic mean of *PR* and *SE*. Finally, the general classifier performance of the was also assessed by receiver-operating-characteristic (ROC) curve analysis; the area under the curve (*AUC*) was the metric used to this effect in this case.

4. Results and discussion

Table 5 shows the results yielded by the five different CNN configurations tested in terms of the metrics defined in equation (12). Attending to performance of the classifiers specifically on the *olive* class, the best *PR* behaviour was provided by Inception-ResNetV2, whereas the best *SE* was measured for InceptionV3; however, in this case, this better *SE* performance was achieved at the expense of an outstanding degradation of *PR*. Indeed, the desirable scenario is that offering the higher combined, but also balanced, *PR* and *SE* score. In this sense, Inception-ResNetV2 clearly offered the best solution, as produced the highest F_1 score of all, showing comparable *PR* and *SE* values.

Table 5

Results of the methodology for detecting olive fruits in olive-tree images, detailed per CNN tried and metric defined in equation (12).

	Metric					
CNN architecture	*SE*	*PR*	*SP*	*ACC*	F_1	*AUC*
AlexNet	0.8744	0.7113	0.9790	0.9731	0.7845	0.9875
VGG19	0.8801	0.7005	0.9777	0.9722	0.7801	0.9877
InceptionV3	0.8943	0.7579	0.9831	0.9776	0.8205	0.9888
ResNet-50	0.8763	0.7329	0.9811	0.9752	0.7982	0.9857
Inception-ResNetV2	0.8313	0.8480	0.9912	0.9822	0.8396	0.9903

The study described in this paper challenged the identification of olive fruits in any case of appearance. The great variability in terms of size, degree of occlusion, and other, present in the training and validation instances, surely increased modelling complexity, and may be the reason behind VGG19 and AlexNet offered the worst performance. Indeed, these architectures are considerably less deep than the other, thus probably being rather limited to capture the great visual variability considered. Besides deepness and regarding olive-size variability, the Inception

approach of including convolutions of different sizes in the same layer also seemed to favour fruit recognition. Certainly, Inception-ResNetV2 and InceptionV3 yielded the first and second best F_1 score, respectively, being additionally the former surely favoured by the integration of the residues formalised by the ResNet philosophy.

All CNNs were configured to have two outputs, one providing the probability for an instance of being *olive* and the other of being *other*; by default, the instance was assigned to the class scored with a probability higher or equal to 0.5. If only the *olive* output is considered, and the instances are assigned to the *olive* class only when the probability is higher or equal to a given threshold *Th*, or to the *other* class when the probability is lower than *Th*, then the CNN can be seen as a classical binary classifier and thus, further analysis can be faced. Under this precept, the performance as a binary classifier of all CNNs was excellent, as drawn ROC curves gave *AUC* values above 0.98 in all cases; furthermore, *ACC* figures exceeded 0.97. Fig. 8 illustrates the ROC curve produced by Inception-ResNetV2, which slightly outperformed the other architectures according to these two metrics. Notwithstanding, note that conclusions based on these findings should be complemented with those deriving from the results discussed above specially for the case of this work, where the *olive* and *other* classes are so significatively unbalanced in terms of number of instances (16,699 vs 283,247).

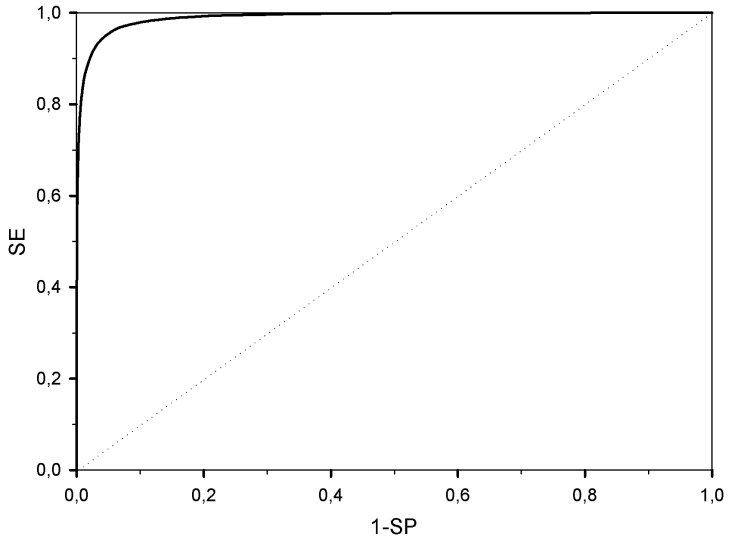

Fig. 8.- ROC curve provided by Inception-ResNetV2. The measured *AUC* was 0.9903.

Based on the performed ROC analysis, F_1 was calculated for every *Th* value from 0 to 1, thus registering the results included in Table 6. As it can be checked, the maximum F_1 was achieved for *Th* values higher than 0.5 in all cases. Notwithstanding, maybe the most remarkable finding of this analysis is that this maximum was achieved with probabilities greater than 0.97, excepting for the case of Inception-ResNetV2, for which the threshold resulted to be 0.6284. Consequently, it can be concluded that Inception-ResNetV2 was able to acquire considerably better 'understanding' about the *olive* and *other* class, compared to AlexNet, VGG19, InceptionV3 and ResNet-50.

Table 6

Analysis of the value of the threshold *Th* to obtain the maximum F_1 response from every CNN.

	Threshold	Metric		
CNN architecture	*Th*	*SE*	*PR*	F_1
AlexNet	0.9970	0.7899	0.8481	0.8180
VGG19	0.9943	0.7928	0.8440	0.8176
InceptionV3	0.9725	0.7949	0.8880	0.8389
ResNet-50	0.9901	0.7893	0.8620	0.8240
Inception-ResNetV2	0.6284	0.8142	0.8678	0.8401

With respect to comparison to other methods outlined in the introduction of the paper, note that none of them faced the identification of olive fruits in olive-tree images taken in the field. Certainly, Ponce et al. (2018), Ponce et al. (2019), Diaz et al. (2004) and Puerto et al. (2015), analysed segmented olive fruits photographed under laboratory conditions to evaluate a variety of characteristics. Conversely, Gatica et al. (2013) developed an image analysis algorithm able to identify olive fruits present in tree brunches cut prior to be photographed over a white homogeneous background. Consequently, the fundamental differences in terms of

463 experimental conditions and approach of these methods with respect to the one presented
464 here, make inviable their scientific and rigorous comparison.
465 Despite the registered and discussed positive results, two strategies were identified as being
466 promising to even improve them in the future. On the one hand, by visually inspecting *FN*
467 instances, a considerably amount of them were identified to occur in peripheral zones of the
468 trees, where the illumination was deficient, which resulted in low-contrasted olive fruits with
469 respect to their surroundings. Consequently, it is reasonably to expect performance improving
470 by designing a system able to produce a more homogeneous canopy illumination. On the other
471 hand, given the described great variability of olive fruits in terms of size and appearance, it is
472 also plausible to expect a general performance improving by drastically increasing the training
473 set with new olive-fruit examples, not computed with data augmentation techniques.
474 The methodology presented is part of a wider solution, currently under investigation, to build a
475 comprehensive and fully automated system for the early olive-fruit yield estimation. In such a
476 system, a robotic platform will autonomously scout the olive orchard, taking images of the olive
477 trees from a lateral view. Then, the described and analysed image analysis algorithm will identify
478 the olive fruits visible in the images, being this information modelled to generate the yield
479 estimations.
480 Super intensive olive orchards are in the focus of interest as the cultivation regime of the future
481 for a variety of reasons, including optimised usage of water or higher productivity, among other.
482 This cultivation also favours automation, and provides an optimum scenario for the
483 comprehensive outlined system, including the methodology discussed in this paper. Indeed,
484 under the super intensive scheme, olive trees are densely cultivated in rows and trained onto a
485 vertical shoot positioned (VSP) trellis system, thus producing a relatively flat and continuous
486 canopy. This ‘flat’ canopy favours its homogeneous illumination and reduces variability in terms
487 of size appearance of olive fruits in images which, as discussed above, provide a more favourable

scenario for the developed methodology. These hypotheses will be analysed and evaluated in the future.

5. Conclusion

Within a frame of global food demand increasing and environmental threat, current systems for early yield estimation of agricultural crops have been found to require a deep revision. To the best of the authors' knowledge, this paper presents the first methodology capable of identifying olive fruits visible in images covering entire olive trees, as a fundamental part of a comprehensive system for early yield estimation of olive orchards. The creation of a specific database, OLIVEnet, publicly put at the disposal of the scientific community to favour further investigation in the subject, is also a novel contribution of this work.

As a first step, the proposal consists in a pre-processing aimed at improving starting image conditions, reducing the potential search space in a magnitude of 10^3, and building an optimum image configuration to empower the performance of a CNN for recognising olives. Then, five of the most successful and recognised CNN configurations were tested, being Inception-ResNetV2 the one clearly showing the best performance and behaviour. Indeed, on a validation set built with 299,946 *olive* and *other* instances, the methodology including this net, correctly identified 83.13% (sensitivity, *SE*) of *olive* instances, with 84.80% of precision (*PR*), and correctly classified 99.12% (*SP*) of *other* instances; measured accuracy (*ACC*) and F_1 Score (F_1) were 0.9822 and 0.8396, respectively. Contrary, the other CNNs tested provided less balanced results, as most of them prioritised *SE* at the expense of deteriorating *PR*. Additionally, a ROC analysis provided similar area under the curve values (*AUC*) for all the explored nets. Notwithstanding, all them achieved the maximum F_1 value at class threshold values higher than 0.97, excepting for the case of Inception-ResNetV2, which achieved this maximum at a threshold value of 0.6484. This finding shows evidences that this net was able to acquire considerably better 'understanding' about the characteristics of both classes.

The shown capabilities of the described methodology open the door to continue investigating, under strong signs of viability, towards the implementation of a comprehensive system for the early olive-fruit yield estimation. Notwithstanding, detected weaknesses point out to improve illumination homogeneity and to increase the training set, which will be addressed with priority. Regarding future work, it will be explored the application of the methodology to images acquired in super intensive orchards, given the existing great cultivated areas under this regime, and its expected prevalence in the future. The characteristics of this cultivation regime, a-priori represents a more favourable scenario for the presented methodology, hypothesis which will be analysed and contrasted. Finally, it will also be conducted the analysis, design and execution of field experiments aimed at acquiring yield data allowing the development of estimation models, by correlating this information to that generated by the present proposal.

Aknowledgments:

Funding: This work was supported by the INTERREG Cooperation Program V-A SPAIN-PORTUGAL (POCTEP) 2014–2020 [grant number 0155_TECNOLIVO_6_E].

Authors would also like to thank Cooperativa Virgen de la Oliva for their support by providing their orchards for the experimentation developed in this paper.

References

Aguilera, F., Ruiz-Valenzuela, L. (2014). Forecasting olive crop yields based on long-term aerobiological data series and bioclimatic conditions for the southern Iberian Peninsula. *Journal of Agricultural Research*, 12(1), 215-224.

Aquino, A., Diago, M.P., Millán, B., Tardáguila, J. (2017). A new methodology for estimating the grapevine-berry number per cluster using image analysis. *Biosystems Engineering*, 156, 80-95.

535 Aquino, A., Millan, B., Diago, M.P., Tardaguila, J. (2018). Automated early yield prediction in
536 vineyards from on-the-go image acquisition. *Computers and Electronics in Agriculture*, 144, 26-
537 36.

538 Bac, C.W., Hemming, J., van Henten, E.J. (2013). Robust pixel-based classification of obstacles
539 for robotic harvesting of sweet-pepper. *Computers and Electronics in Agriculture*, 96, 148–162.

540 Bargoti, S., Underwood, J.P. (2017a). Image segmentation for fruit detection and yield
541 estimation in apple orchards. *Journal of Field Robotics*, 34(6), 1039-1060.

542 Bargoti, S., Underwood, J. (2017b). Deep fruit detection in orchards. In *IEEE International*
543 *Conference on Robotics and Automation (ICRA)*, 3626-3633.

544 Connolly, C., Fliess, T. (1997). A study of efficiency and accuracy in the transformation from RGB
545 to CIELAB color space. *IEEE Transactions on Image Processing*, 6, 1046–1047.

546 Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical
547 image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

548 Diaz, R., Gil, L., Serrano, C., Blasco, M., Moltó, E., Blasco, J. (2004). Comparison of three
549 algorithms in the classification of table olives by means of computer vision. *Journal of Food*
550 *Engineering*, 61(1), 101–107.

551 Duda, R.O., Hart, P.E., Stork, D.G. (2012). *Pattern classification, second ed.* New York, USA: John
552 Wiley & Sons.

553 European Commission–Agriculture and Rural Development. (2011). Agricultural Markets Briefs–
554 Brief 1: High commodity price and volatility: what lies behind the roller coaster ride?. Resource
555 document.

556 http://ec.europa.eu/agriculture/analysis/tradepol/commodityprices/market-briefs/01_en.pdf.
557 Accessed 24 April 2020.

558 Fabio, O., Carlo, S., Tommaso, B., Luigia, R., Bruno, R., Marco, F. (2010). Yield modelling in a
559 Mediterranean species utilizing cause–effect relationships between temperature forcing and
560 biological processes. *Scientia Horticulturae*, 123(3), 412-417.
561 Font, D., Tresanchez, M., Martínez, D., Moreno, J., Clotet, E., Palacín, J. (2015). Vineyard yield
562 estimation based on the analysis of high resolution images obtained with artificial illumination
563 at night. *Sensors*, 15, 8284–8301.
564 Food and Agriculture Organization of the United Nations (FAOSTAT). (2018). Olive yield statistics
565 for 2018. Resource document.
566 http://www.fao.org/faostat/en/#home. Accessed 24 April 2020.
567 Fornaciari, M., Orlandi, F., Romano, B. (2005). Yield forecasting for olive trees. *Agronomy*
568 *Journal*, 97(6), 1537-1542.
569 Galán, C., García-Mozo, H., Vázquez, L., Ruiz, L., Díaz de la Guardia, C., Domínguez-Vilches, E.
570 (2008). Modeling Olive Crop Yield in Andalusia, Spain. *Agronomy Journal*, 100, 98-104.
571 Galán, C., Vázquez, L., Garcia-Mozo, H., Dominguez, E. (2004). Forecasting olive (Olea europaea)
572 crop yield based on pollen emission. *Field Crops Research*, 86(1), 43-51.
573 Gatica, G., Best, S., Ceroni, J., Lefranc, G. (2013). Olive fruits recognition using neural networks.
574 *Procedia Computer Science*, 17, 412-419.
575 Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, L., Wang, G.,
576 Cai, J., Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*,
577 77, 354–377.
578 He, K., Zhang, X., Ren, S., Sun, J. (2015). Deep residual learning for image recognition. In *IEEE*
579 *Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778.
580 Hung, C., Nieto, J., Taylor, Z., Underwood, J., Sukkarieh, S. (2013). Orchard fruit segmentation
581 using multi-spectral feature learning. In *IEEE/RSJ International Conference on Intelligent Robots*
582 *and Systems*, 5314-5320.

583 Krizhevsky, A., Hinton, G. (2009). Learning Multiple Layers of Features from Tiny Images.
584 *University of Toronto*, 1(4), 7.
585 Krizhevsky, A., Sutskever, I., Hinton, G.E. (2012). Imagenet classification with deep convolutional
586 neural networks. In *International Conference on Neural Information Processing Systems*, 1097-
587 1105.
588 Kumar, G., Bhatia, P.K. (2014). A Detailed Review of Feature Extraction in Image Processing
589 Systems. In *Fourth International Conference on Advanced Computing & Communication*
590 *Technologies*, 5–12.
591 Lecun, Y., Bengio, Y., Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.
592 Liu, S., Marden, S., Whitty, M. (2013). Towards automated yield estimation in viticulture. In
593 *Proceedings of the Australasian Conference on Robotics and Automation*.
594 Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., Alsaadi, F.E. (2016). A survey of deep neural network
595 architectures and their applications. *Neurocomputing*, 234, 11–26.
596 Millan, B., Velasco-Forero, S., Aquino, A., Tardaguila, J. (2018). On-the-Go Grapevine Yield
597 Estimation Using Image Analysis and Boolean Model. *Journal of Sensors*, 2018(Article ID
598 9634752), 1-14.
599 Minero, F.J.G., Candau, P., Morales, J., Tomas, C. (1998). Forecasting olive crop production based
600 on ten consecutive years of monitoring airborne pollen in Andalusia (southern Spain).
601 *Agriculture Ecosystems & Environment*, 69(3), 201-215.
602 Nguyen, T.T., Vandevoorde, K., Wouters, N., Kayacan, E., De Baerdemaeker, J.G., Saeys, W.
603 (2016). Detection of red and bicoloured apples on tree with an RGB-D camera. *Biosystems*
604 *Engineering*, 146, 33-44.
605 Nuske, S., Wilshusen, K., Achar, S., Yoder, L., Narasimhan, S., Singh, S. (2014) Automated visual
606 yield estimation in vineyards. *Journal of Field Robotics*, 31, 837–860.

607 Orlandi, F., Sgromo, C., Bonofiglio, T., Ruga, L., Romano, B., Fornaciari, M. (2010). Yield modelling
608 in a Mediterranean species utilizing cause-effect relationships between temperature forcing and
609 biological processes. *Scientia Horticulturae*, 123(3), 412-417.

610 Oteros, J., Orlandi, F., García-Mozo, H., Aguilera, F., Dhiab, A.B., Bonofiglio, T., Abichou, M.,
611 Ruiz-Valenzuela, L., Mar del Trigo, M., Díaz de la Guardia, C., Domínguez-Vilches, E., Msallem,
612 M., Fornaciari, M., Galán, C. (2014). Better prediction of Mediterranean olive production using
613 pollen-based models. *Agronomy for Sustainable Development*, 34(3), 685-694.

614 Ponce, J.M., Aquino, A., Millan, B., Andújar, J.M. (2018). Olive-fruit mass and size estimation
615 using image analysis and feature modelling. *Sensors* 18(9), 2930, 1-14.

616 Ponce, J.M., Aquino, A., Millan, B., Andújar, J.M. (2019). Automatic Counting and Individual Size
617 and Mass Estimation of Olive-Fruits Through Computer Vision Techniques. *IEEE Access*, 7, 59451-
618 59465.

619 Puerto, D.A., Gila, D.M.M., García, J.G., Ortega, J.G. (2015). Sorting olive batches for the milling
620 process using image processing. *Sensors* 15(7), 15738–15754.

621 Qureshi, W.S., Payne, A., Walsh, K.B., Linker, R., Cohen, O., Dailey, M.N. (2017). Machine vision
622 for counting fruit on mango tree canopies. *Precision Agriculture*, 18(2), 224-244.

623 Shin, H.-C., Roth, H.R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., Summers, R.M.
624 (2016). Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures,
625 Dataset Characteristics and Transfer Learning. *IEEE Transactions on Medical Imaging*, 35(5),
626 1285 – 1298.

627 Simonyan, K., Zisserman, A. (2014). Very deep convolutional networks for large-scale image
628 recognition. In *International Conference on Learning Representations (ICLR)*.

629 Smith, W.J. (2000). *Modern Optical Engineering, third ed.* California, USA: McGraw-Hill.

630 Soille, P. (2004) *Morphological Image Analysis - Principles and Applications, second ed.* Berlin,
631 Germany: Springer – Verlag.

Sonka, M., Hlavac, V., Boyle, R. (2014). *Image processing, analysis, and machine vision, fourth ed.* USA: Cengage Learning.

Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A. (2017). Inception-v4, inception-ResNet and the impact of residual connections on learning. In *31st AAAI Conference on Artificial Intelligence (AAAI-17)*, 4278-4284.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A. (2015). Going deeper with convolutions. In *28th IEEE conference on computer vision and pattern recognition (CVPR)*, 1–9.

Van der Velde, M., Nisini, L. (2019). Performance of the MARS-crop yield forecasting system for the European Union: Assessing accuracy, in-season, and year-to-year improvements from 1993 to 2015. *Agricultural Systems*, 168(2019), 203-212.

Vitzrabin, E., Edan, Y. (2016). Adaptive thresholding with fusion using a RGBD sensor for red sweet-pepper detection. *Biosystems Engineering*, 146, 45-56.

Voulodimos, A., Doulamis, N., Doulamis, A., Protopapadakis, E. (2018). Deep Learning for Computer Vision: A Brief Review. *Computational Intelligence and Neuroscience*, 2018(Article ID 7068349), 1-13.

Yamamoto, K., Guo, W., Yoshioka, Y., Ninomiya, S. (2014). On plant detection of intact tomato fruits using image analysis and machine learning methods. *Sensors*, 14(7), 12191–12206.

Zhao, Y., Gong, L., Zhou, B., Huang, Y., Liu, C. (2016). Detecting tomatoes in greenhouse scenes by combining AdaBoost classifier and colour analysis. *Biosystems Engineering*, 148, 127-137.