Application of deep convolutional neural networks for the detection of anthracnose in olives using VIS/NIR hyperspectral images

Antonio Fazari*[1)], Oscar J. Pellicer-Valero*[2)], Juan Gómez-Sanchıs[2)], Bruno Bernardi[1)], Sergio Cubero[3)], Souraya Benalia[1)], Giuseppe Zimbalatti[1)], Jose Blasco[3)]

[1)] Dipartimento di Agraria. Università degli Studi Mediterranea di Reggio Calabria, Loc. Feo di Vito, 89122 Reggio Calabria, Italy. E-mail: bruno.bernardi@unirc.it, soraya.benalia@unirc.it, gzimbalatti@unirc.it, antonio.fazari@unirc.it

[2)]Intelligent Data Analysis Laboratory, Department of Electronic Engineering, ETSE (Engineering School), Universitat de València (UV), Av. Universitat, sn, 46100 Bujassot, Valencia, Spain. E-mail: Oscar.Pellicer@uv.es, Juan.Gomez-Sanchis@uv.es

[3)]Centro de Agroingeniería, Instituto Valenciano de Investigaciones Agrarias (IVIA). Carretera CV-315, Km 10.7, 46113 Moncada, Spain. E-mail: blasco_josiva@gva.es, cubero_ser@gva.es

*These authors contributed equally to the work

Abstract

Anthracnose is one of the primary diseases that affect olive production before and after harvest, causing severe damage and economic losses. The objective of this work is to detect this disease in the early stages, using hyperspectral images and advanced modelling techniques of Deep Learning (DL) and convolutional neural networks (CNN). The olives were artificially inoculated with the fungus. Hyperspectral images (450-1050 nm) of each olive were acquired until visual symptoms of the disease were observed, in some cases up to 9 days. The olives were classified into two classes: control, inoculated with water, and fungi composed of olives inoculated with the fungus. The ResNet101 architecture was chosen and adapted to process 61-band hyperspectral images with only two classes. The result showed that the applied model is very effective in detecting infected olives since the sensitivity of the method was very high from the beginning (85% on day 3 and 100% onwards). From a commercial point of view, these results align with the need to detect the maximum number of infected fruits.

Keywords: *Olea europaea* L.; quality inspection; fungi; computer vision; spectral imaging

1. INTRODUCTION

Olive (*Olea europaea* L.) cultivation represents a key sector for the economy of the entire Mediterranean basin, where this crop extends over 10.6 million hectares in the world, of which 24% is grown in Spain and 11% in Italy (FAOSTAT, 2018). Olive oil is a highly valued product that plays a key role in the Mediterranean diet. In this context, the quality of drupes and hence the olive oil is deeply influenced by several factors, such as pedoclimatic and weather conditions, the cultivar, agronomic practices or postharvest handling. Recently, phenomena associated with climate change have favoured the spread of diseases like anthracnose. Several species of Colletotrichum cause this disease, mostly grouped in the *C. acutatum* species complex. Anthracnose is one of the most critical and dangerous fungal diseases of olive production worldwide. The anthracnose of the olive is known as an epidemic since 1898 when it was

described in Portugal by Almeida (1899) and named “gaffa”. Currently, it is known as “lebbra” in Italy and as “aceituna jabonosa” in Spain, where the term “repilos” is commonly used to indicate collectively anthracnose, peacock spot and cercosporiosis (Cacciola et al., 2012).

The symptoms of anthracnose appear as circular necrotic lesions, initially depressed, of ochre or brown colour. They can occur in the apical zone but often also near the peduncle, which later converges with each other with the consequent rotting of the pericarp (Trapero Casas, 2003). In immature green fruits, the infection remains inactive until ripening, and the latency period is inversely correlated with the time of infection, which favours epidemics in autumn (Moral and Trapero, 2009). However, in ripe fruits, the presence of browning of the pulp can also be observed before the appearance of symptoms (Talhinhas et al., 2005). If the harvested fruits show an incidence of olive anthracnose higher than 5%, the quality of the olive oil produced degrades (Moral et al., 2014), causing sensory defects, such as "soil" and "mould" flavours, and chemical degradation (Carvalho et al., 2008).

Visual inspection is the traditional method of detecting and identifying diseases in the field, but it requires extensive experience. Currently, there are techniques to detect pathogen-specific antigens or oligonucleotides, such as immunoassays (e.g., enzyme-linked immunosorbent assay, ELISA), polymerase chain reaction (PCR) or ribonucleic acid (RNA) sequencing. But these methods are expensive, time-consuming, and require highly trained human resources. In addition, they are not repeatable as they are performed on representative samples that cannot guarantee the quality of the entire batch, leading to inaccurate quality assessment. Artificial vision systems have been developed in recent years as an alternative to traditional postharvest visual inspection (Cubero et al., 2011), including the inspection of table olives (Díaz et al., 2000 and 2004). No imaging techniques have been addressed to date to detect symptoms of anthracnose in olives. The only automated sensor reported has been an artificial nose to detect odour compounds in olive oil-related to diseased olives during the producing process. Among the imaging techniques, hyperspectral imaging is one of the most widely used and promising as a non-destructive postharvest inspection method (Lorente et al., 2012), especially when combined with machine learning techniques (Munera et al., 2021).

Standard vision systems for fruit sorting are typically based on colour images. Therefore, they cannot get information on the external or internal composition of the products or detect some defects or alterations whose colour is close to the colour of the sound skin, but that involve chemical changes in the skin. In this regard, hyperspectral imaging systems allow detecting diseases in early stages or even when they are still invisible to the human eye (Gómez-Sanchis et al., 2014). The causative agents of anthracnose in tea leaves were studied by Yuan et al., (2019) using this technology. They distinguished healthy tea leaves from infected ones using three bands (542, 686, and 754 nm) that characterised the infected leaves. Yeh et al., (2016) demonstrated that the different stages of infection of strawberry foliar anthracnose caused by *Colletotrichum gloeosporioides* could be identified by analysing hyperspectral images using four wavelengths (551, 706, 750 and 914 nm). Sun et al., (2015), in a study related to the growth and discrimination of *Botrytis cinerea*, *Rhizopus stolonifer* and *Colletotrichum acutatum* through hyperspectral images, identified three absorption peaks corresponding to 560 nm, 770 nm and 970 nm. They found changes related to the spectral values of *C. acutatum* from 24 h to 36 h of growth. The changes followed the development of the fungi, so the data obtained could be helpful to predict the presence of *C. acutatum* using hyperspectral imaging. Lu et al., (2017) investigated the feasibility of detecting anthracnose crown rot in strawberry in its initial stage under field conditions using spectroscopy. The results showed that the reflectance was higher in infected leaves than in healthy and asymptomatic leaves, particularly in the ranges of 400 to 520 nm and 580 to 697 nm. However, they noted that this trend reversed from 520 to 580 nm.

One of the hyperspectral image problems is the enormous amount of data it generates (Gómez-Sanchis et al., 2013). Therefore, powerful feature extraction methods are required to obtain accurate information. Recently, techniques based on Deep Learning (DL) and convolutional neural networks (CNN) have emerged as a powerful tool to process large amounts of data. The study of neural networks originally began in the 1940s (McCulloch and Pitts, 1943), and

Fukushima (1980) and LeCun et al., (1989) introduced the first CNNs. Deep CNNs (DCNNs) are flexible algorithms that have already been used successfully in fresh fruit inspection problems (Ashraf et al., 2019; Steinbrener et al., 2019). Sladojevic et al., (2016) proposed a model to detect different types of diseased peach, cherry, pear, apple, and vine leaves using CNN, achieving an accuracy between 91% and 98%. A similar model was used by Fan et al., (2020) to detect defective apples in-line with a success of 92%, and Chen et al (2020) to detect defective oranges with a 93.6% success. Nasiri et al., (2019) used a DCNN constructed from a VGG-16 architecture to classify date fruit into four classes, including defective dates, with success based on accuracy, precision, specificity and sensitivity between 96 and 99%. Using DCNN, Sharada et al., (2016), used a public data set of 54306 images of 14 crops and 26 diseases to develop a method for detecting plant diseases, reaching an accuracy between 98% and 99%. Wallelign et al., (2018) designed a model to identify soybean plant diseases, achieving 99% accuracy. Ferentinos (2018), used a data set of 25 different plants and 58 diseases, reaching a precision of 99.5%. This work focuses on the early detection of anthracnose in olives using visible (VIS) and near-infrared (NIR) hyperspectral images, combined with DL and CNN techniques. Xiao et al., (2021) used a CNN ResNet50 model with 20 epochs for 1306 feature images to discriminate among various diseases in strawberry plants, including anthracnose. Based on the ability of hyperspectral imaging systems to obtain information about the non-visible properties of the fruit skin, the objective was to detect the infection of *C. gloeosporioides* (anthracnose) in olives in the earliest possible stages, using both visible (VIS) and near-infrared (NIR) hyperspectral imaging combined with DL and CNN techniques.

2. MATERIALS AND METHODS

2.1. Olive inoculation

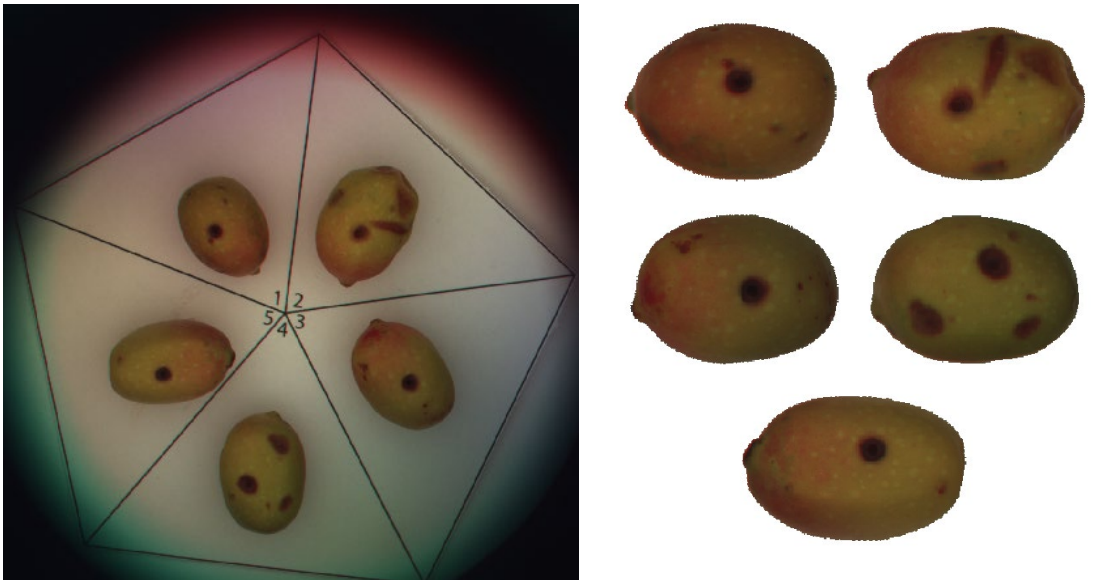
A total of 250 olives free of external defects were used for this experiment, of which 80% were inoculated with *C. acutatum.* The remaining 20% were inoculated with water as a control to know if the injury caused by the inoculation process had any effect in the results of detection of the anthracnose (Lorente et al., 2015). The fungus was isolated from rotten olives harvested in Calabria, southern Italy. The fungus *C. acutatum* was identified according to the methodology described by Schena et al., (2014). To produce the inoculum, a purified conidia suspension of the pathogen was plated on potato dextrose agar (PDA) (Sigma-Aldrich, MI, USA) and incubated at 25 °C for 7-10 days. Later, to prepare the inoculum, the grown conidia of *C. acutatum* were harvested with a spatula, suspended in sterile distilled water, filtered through a double layer of sterile muslin cloth, and mixed evenly. The conidia concentration was determined using a hemocytometer chamber and diluted to obtain stock solutions containing 105 conidia/ml (Pangallo et al., 2017).

Olives cv. 'Carolea', were collected in a commercial farm located in Reggio Calabria, in southern Italy. The samples were cleaned by immersing them in a 2% sodium hypochlorite solution for approximately 2 minutes. They were then rinsed with water and allowed to dry. The drupes were wounded (2 mm wide and 2 mm deep) with a needle in their equatorial region and subsequently inoculated with 10 microliters of the suspension described above. The inoculated drupes were kept at room temperature (20-22 °C) and 100% relative humidity in closed plastic containers.

2.2. Image acquisition and dataset

The hyperspectral imaging system consisted of an industrial camera (CoolSNAP ES, Photometrics, AZ, USA) coupled to two liquid crystal tunable filters (LCTF) (Varispec VIS-07 and NIR-07, Cambridge Research & Instrumentation, Inc., MA, USA). The camera was configured to acquire images with 1392 × 1040 pixels with a spatial resolution of 0.14 mm/pixel. The working spectral range was defined between 450 nm and 1050 nm, capturing images every 10 nm. Thus, hypercubes with 61 images were captured.

An advantage of using LCTF is that the exposure time can be set independently for each band. Therefore, to optimise the dynamic range of the camera and correct the spectral sensitivity of the different elements of the system (lamps, filters and camera), the integration time was established for each band in a manner. This time was determined so that 90% of the average reflectance of a standard white reference (Spectralon 99%, Labsphere, Inc, NH, USA) was obtained for each band. Diffuse lighting was provided by 12 halogen bulbs (37 W) (Eurostar IR Halogen MR16. Ushio America, Inc., CA, USA) supplied with direct current (12 V) and arranged equidistant from each other within a hemispherical diffuser of aluminium.



a) b)

Figure 1: Images of the olives (shown using the RGB bands) as they were acquired (a) and after the pre-processing (b).

Images of each olive were captured daily from the third day after inoculation until the visual symptoms of rot evolved (in some cases, up to 9 days). Therefore, a variable number of images of each olive was captured depending on the evolution of the infection, obtaining a total of 610 hyperspectral images: 145 at day 3, 155 at day 4, and 310 at days 5-9. Rather than studying the evolution of the defect, the purpose was to obtain images with different development stages of the defect.

The olives were arranged in a pentagonal pattern to be all illuminated in the same way (Figure 1a). To build the data set for the CNN, each olive was automatically identified in the image, extracted and saved independently (Figure 1b). First, the 480 nm band was used to obtain a binary mask and remove the background since the contrast between the olives and the background was maximum in this band. The olives were then individualised and resized to a size of 224x224 pixels, which is typical for many CNN architectures (He et al., 2016). Finally, the hyperspectral image of each olive was stored in Python's pickle format, which is a simple and efficient way of serialisation in a Python-oriented pipeline. Hyperspectral images were corrected and transformed to relative reflectance using white (Spectralon 99%) and dark references following the calibration process described in Geladi (2007).

2.3. Train, test and validation subsets

The olives were divided into two classes. The class water corresponded to control olives that were inoculated with water (20%), and the fungi class composed of the olives inoculated with the fungus (80%). The greater number of inoculated olives was due to the main objective of the work, which was to investigate the capacity of the system to detect this disease early. To evaluate the models, the images were divided into three subsets for training (60%), validation (20%), and independent testing (20%). The training set was used to optimise the CNN weights through the Stochastic Gradient Descent (SGD) (Ruder, 2016), which is an iterative algorithm that calculates the classification error (or loss) in each iteration on a subset of images, and the weights are then updated to reduce this error. This method improves the performance of the classification throughout the process until some stopping criterion is met (for example, the error is minimal, or a maximum number of iterations is reached). The validation set was employed to manually optimise design decisions related to the CNN architecture and the training procedure. Only those changes that improved validation performance were preserved. Finally, the test set was used to assess the generalizability of the model, providing an unbiased estimate of its performance on data not seen during training or validation.

These three sets were obtained by taking three olives for training, one for validation, and one for the independent test from each of the five olives acquired per shot (see Figure 1), such that the olive in the same position always belonged to the same set. Otherwise, the validation and test sets could contain images of olives also present in the training set. Hence, the model could have learned to recognise them during this process, thus compromising the validity of the results. Furthermore, the relative proportion between the two classes (20% control, 80% fungus) was conserved in each of the sets, as well as the ratio of olives captured at each date.

In addition to the described main validation scheme, 5-fold cross-validation (CV) was employed to prove the generality of the final system further after the architecture and hyperparameters were considered final. In this scenario, five different models are trained using four out of the five olives from each shot, while the fifth one is left out for testing (a different one for each of the models).

2.4. CNN architecture

Standard classification CNN's can be understood as the concatenation of a feature extractor block with a classification head. The feature extractor block processes an input image through a series of convolutions, pooling operations, and non-linear activation functions, producing a feature map at each step. The final feature map is projected onto a feature vector, which can be understood as a highly compressed representation of the input image, containing only important semantic features. This vector is supplied to the classification head, which is a fully connected classical neural network that produces a vector of probabilities of the input image belonging to each class. As an example, Figure 2 shows the architecture of the well-known CNN VGG-16 architecture (Simonyan and Zisserman, 2015).

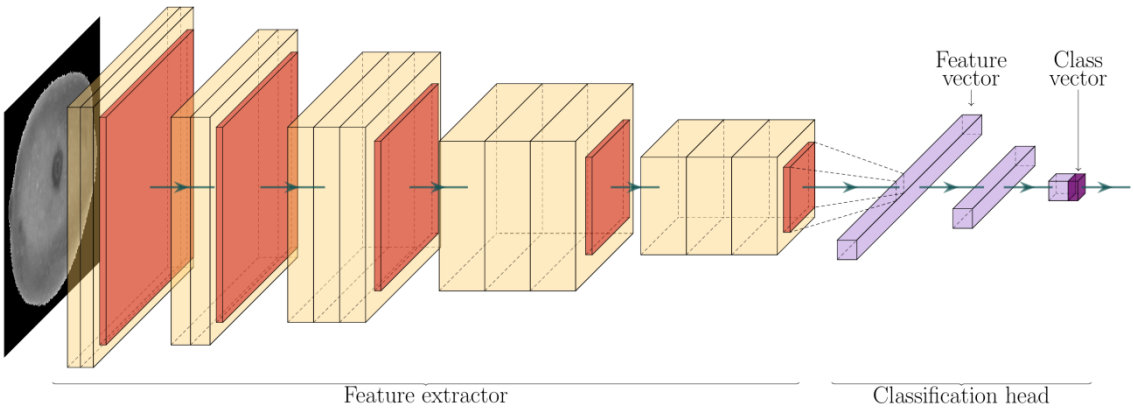


Figure 2: Scheme of the VGG-16 architecture (Simonyan and Zisserman, 2015).

After testing several competing CNN architectures, such as VGG-16, ResNet50 (He et al., 2016) and Densenet121 (Huang et al., 2016), the ResNet101 (He et al., 2016) architecture was finally chosen because it produced the best results over the validation set and represents a good balance

between discriminatory power, parameter efficiency, simplicity and learning transferability. ResNet101 has a feature extractor that consists of a stack of 101 convolutions structured in 33 blocks of three convolutions each, with a residual connection that joins the beginning of each block to its end. It uses the rectified linear activation function (ReLU) (Eckle and Schmidt-Hieber, 2019).

At the end of the CNN, the classification head processes the resulting feature map through a fully connected layer with as many outputs as there are classes, using the normalised exponential function (softmax) as an activation function (Bishop, 2006), thus producing the final predictions.

The original implementation of ResNet101 was designed for the ImageNet dataset (Russakovsky et al., 2015), which contains more than one million colour images (3 bands) that belong to 1000 different categories. Therefore, some modifications were necessary to adapt this network to process 61-band hyperspectral images with only two classes. To address the large number of input bands (known as channels in the context of CNNs), a trainable linear transformation was added at the beginning of the CNN. This was implemented as a convolution with a kernel of size (1, 1), which performs pixel compression of the 61 channels of the hyperspectral images captured in only the three with which this network works. It is important to note that this reduction was not based on selecting important bands since the three resulting channels were not particular bands but a projection of the information contained in the original 61 bands. This concept is similar to using Principal Component Analysis to analyse the 61 bands and obtain the three most important components, except that the transformation is learned during the CNN training process.

Similarly, the classification header had to be redesigned for a binary classification problem (sound or infected). Therefore, the last activation map was linearly compressed from 2048 to 64 channels and then collapsed into one dimension (flattened) to obtain the feature vector. The actual classification head was replaced by a fully connected network with a single hidden layer of 16 neurons.

The final architecture can be seen in Figure 3, where the original ResNet101 feature extractor remains unchanged, and the rest of the implementation has been adapted to our problem. As a summary, each hyperspectral image of 61 bands from each olive was rescaled to a size of 224x224 and transformed into a three-channel (not bands) image through a linear transformation pixel-wise (equivalent to a convolution with a kernel size 1x1). The three channels were used as input in the trained CNN ResNet101, obtaining as output a 7x7x2048 activation map. This map was compressed from 2048 channels to only 64 channels (using another convolution with a kernel size 1x1) and flattened to obtain the vector of features associated with the image. Finally, two other layers (the first with 16 units and the ReLU activation function, and the second with two units and the softmax activation function) formed the classification head that obtained the probability of the olive being healthy or infected.

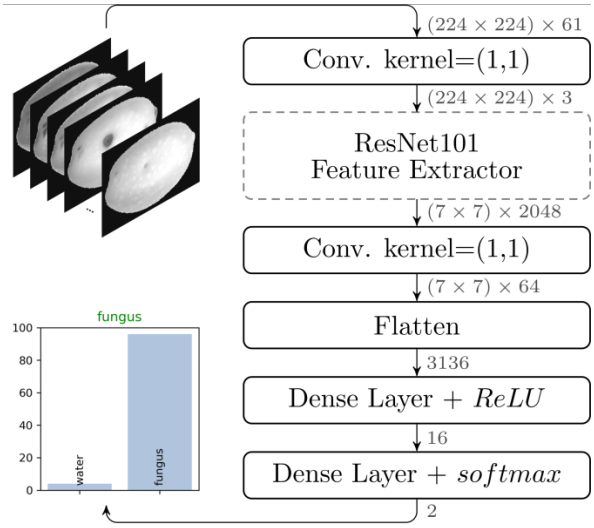


Figure 3: Architecture proposed for the model created.

2.5. Addressing overfitting

Effective CNN training is challenging when the data set is relatively small, as in this case. Such a complex model trained with little data generally performs well with the training set but lacks generalizability, thus exhibiting poor results in the test and validation sets, a problem known as overfitting. In this work, two standard techniques were employed to address this problem: transfer learning and on-line data augmentation.

2.5.1. Transfer learning

Transfer learning (Yosinski et al., 2014) consists of using a CNN that has been previously trained on a large dataset (such as ImageNet) and applying it to a new (often smaller) dataset. Filters learned by any CNN (especially those in the first layers) are extremely general and therefore useful for various image classification tasks. Taking advantage of this technique, the proposed CNN did not need to train from scratch, helping it converge to a local minimum despite the relatively small data set.

There are two main approaches to transferring learning. The first consists of using the function extractor of an already trained CNN as a function extractor for a new problem, without any parameter retraining. Then any standard classifier can be trained directly on the resulting feature vectors, replacing the original classification head. This can be effective when the domain that the original CNN was trained in is very similar to that of the new problem. In our case, however, there was a major domain shift, as the hyperspectral olive images were very different from the images included in ImageNet (everyday scenes). Therefore, this approach would have been unsuccessful for our application.

The second approach, known as fine-tuning, uses the weights of the previously trained CNN as an initialisation for the new one and then slowly retrains it so that the feature extractor can adapt to the new domain. This strategy was adopted for our model, which employs a ResNet101 feature extractor pre-trained on ImageNet, that was slowly retrained using the new olive images. Section 2.6 illustrates the training process.

2.5.2. On-line data augmentation

Augmentation of on-line data (Pérez and Wang, 2017) is a technique that is commonly used to artificially increase the amount and variability of training images when the data set is small, as is our case. Using this method, a series of random transformations were applied sequentially to each image before they were fed into the CNN for training: flip along the x-axis with probability 1/2, alter the brightness in a range of ± 8%, alter the contrast in a range of ± 40%, add Gaussian noise with a standard deviation of 0.02, rotate by an angle in the range [0, 2π], shift both axes by a maximum of ± 8%, and scale the image in the range [0.7, 1.02] times. Figure 4 shows the results of this process in some example olives.

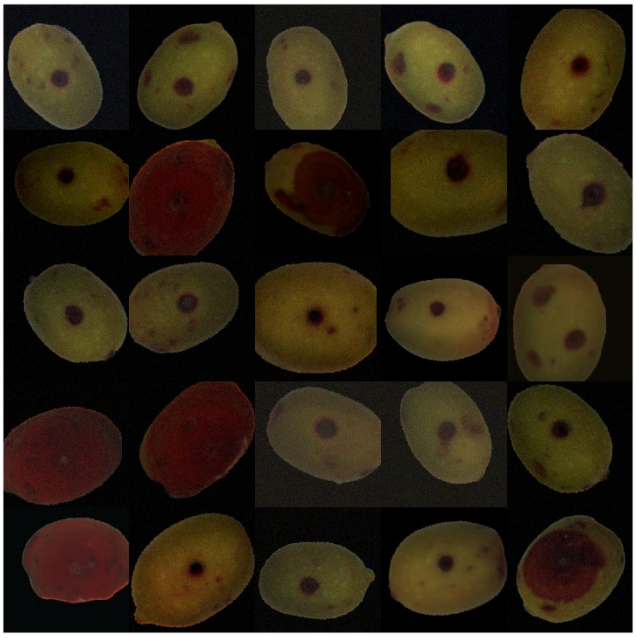


Figure 4: Some samples after applying data augmentation.

2.6. Training schedule

For successful transfer learning, a CNN had to be carefully trained so that the pre-trained weights were not wholly overwritten during the retraining process. For our case, a two-step training process was carried out: first, the weights of the previously trained ResNet101 were locked and only the weights of the newly added layers were updated for 20 epochs, where an epoch is defined as 12 SGD iterations (the amount required to approximately iterate through the whole dataset once). For this, the SGD Adam optimiser (Kingma and Ba, 2015) was used, with a learning rate of 1.67×10^{-4} and a batch size of 32 images.

Second, all weights were unlocked and updated using standard SGD with a momentum of 0.9 at a slower learning rate of 4.18×10^{-5}. For this second phase, the learning rate was automatically reduced by a factor of 0.3 if the loss did not improve in the last 20 epochs in the validation set. Finally, the training stopped when the validation loss had not improved in 45 epochs, which is a technique known as early stopping. This also helped reduce the overfitting problem, as the training process was stopped before the model had a chance to overfit the training dataset. The Python's library Tensorflow 2.0 was used for training the CNN.

In all cases, categorical cross-entropy (Equation 1) was chosen as the loss function since it has some mathematical properties that make it a good choice as a minimisation objective in multi-class classification problems, while accuracy (Equation 2) was used as the primary evaluation metric, due to its superior interpretability.

$$Cross - entropy = -\sum_{i=1}^{N}\sum_{c=1}^{M} y_{i,c}\log\left(\hat{y}_{i,c}\right) \qquad (1)$$

$$Accuracy = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{M} \mathrm{I}\left(y_{i,c} = \hat{y}_{i,c}\right) \qquad (2)$$

Where N is the number of samples, M is the number of classes, y is the ground truth, $\hat{y}$ is the value predicted by the CNN, and $\mathrm{I}: X \rightarrow \{0,1\}$ is the indicator function that takes a value of 1 when the internal condition is fulfilled and 0 otherwise.

2.7. Test Time Augmentation (TTA)

After training the model, TTA (Wang et al., 2018) was used to improve the predictions and performance of the model. TTA consists in combining the predictions on a given image that has been augmented with a random transformation, similar to the augmentations made during training using on-line data augmentation. This allows for a better exploration of the space of possible appearances that a particular olive can take when being photographed, thus improving the robustness and performance of the prediction. Majority voting was used as a combining function, which is a common choice for classification tasks. Each image was augmented 25 times. Thus, when TTA was used to classify a single olive image, it was first transformed in 25 different ways by random rotations, translations, etc. These images were then supplied to the CNN, thus obtaining a total of 25 predictions for the same original image, which were then combined into a single prediction by choosing the most voted class.

3. Results and discussion

3.1. Performance evaluation

The performance of the model was evaluated in terms of accuracy, sensitivity and specificity. The accuracy represents the total fraction of images that were correctly classified. The sensitivity is the fraction of infected olives that were detected, while the specificity is the fraction of healthy olives that were correctly identified as such. For comparison purposes, a Naïve model (not to be confused with a naïve Bayes classifier) that always classifies samples as infected, would achieve only 80% accuracy (due to there being 80% infected samples), 0% specificity (healthy olives would never be identified), and 100% sensitivity (all infected olives would have been detected). Finally, the F1 score is the harmonic mean of precision and recall and provides an alternative measure of classification performance.

The accuracy, sensitivity and specificity for the training, validation and test sets (including all stages of infection) are shown in Table 1. As expected, the model performs better in the training set. However, the difference with the other subsets is relatively small due to the techniques applied to avoid overfitting.

Table 1: Accuracy, sensitivity, specificity and F1 score for train, validation and test subsets.

Subset	Accuracy (%)	Sensitivity (%)	Specificity (%)	F1 (%)
Train	92.67	97.49	73.77	95.49
Validation	92.62	93.75	88.46	95.24
Test	91.80	96.88	73.08	94.90
Naïve	80.00	100	0	88.07

Table 1 shows that the true positive rate (sensitivity) was high in training, validation and test subsets, staying above 90% in all cases. However, the most notable result was the sensitivity of 96.88% in the test set, which demonstrates the high generalizability of the model for detecting anthracnose in olives.

More modest results were observed in terms of specificity in training (73.77%), validation (84.62%) and test (73.08%) sets. However, in most food quality control and inspection problems, high sensitivity rates are more important than specificity since it is more tolerable to over-detect those in poor condition (which implies an economic loss) than to classify infected as healthy (which also means a loss of prestige and the rejection of the product). In this case, an average specificity of around 80% was reached.

Table 2 presents the results from the 5-fold validation scheme. For the first and third folds, the results were similar to those in Table 1. For the rest of the folds, the results were slightly worse. Due to the employed small imbalanced dataset, a specific subset of the data might be significantly better or worse suited for learning, causing these issues.

Table 2: Accuracy, sensitivity, specificity and F1 score for the five different CV test subsets.

Subset	Accuracy (%)	Sensitivity (%)	Specificity (%)	F1 (%)
CV Fold 1 Test	90.98	94.79	76.92	94.30
CV Fold 2 Test	86.89	94.79	57.69	91.92
CV Fold 3 Test	88.52	98.96	50.00	93.14
CV Fold 4 Test	84.43	97.92	34.62	90.82
CV Fold 5 Test	80.99	100	11.54	89.20

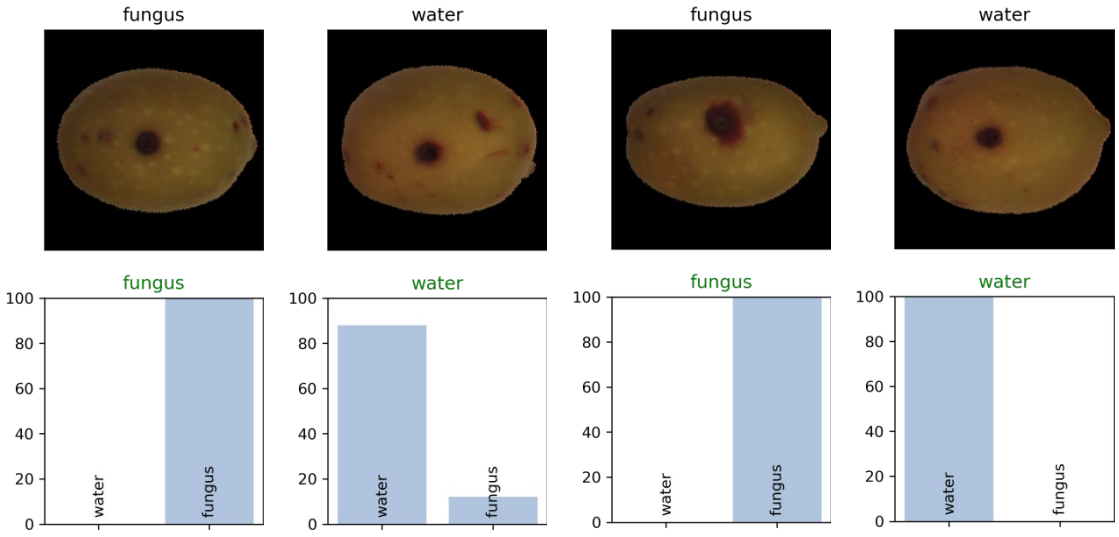
The acquisition of images in different stages of evolution of the defects was carried out only to obtain a broader sample of defects. However, this allows for information on the performance of the model to be obtained according to the evolution of the infection.

Table 3 shows the accuracy, sensitivity, specificity and F1 score obtained for the olive images from the test set that were acquired on days 3, 4 and 5 to 9 after inoculation. As can be seen, as the disease evolved, accuracy, sensitivity, and specificity increased. On days 3 and 4, the infection had already started to develop. However, the visual appearance of the olive still appeared like the damage caused by the inoculation. The specificity was only 50% on days 3 and 4, but the model was effective in detecting the infected olives since the sensitivity was high from the beginning (85% on day 3 and 100% onwards). For comparison purposes, results achieved training the model using only RGB bands are included in the table.

Table 3: Accuracy, sensitivity, and specificity achieved using the independent test set for samples collected on days 3, 4 and 5 to 9 after inoculation, using the full spectrum and only the RGB bands.

Bands	Days after inoculation	Accuracy (%)	Sensitivity (%)	Specificity (%)	F1 (%)
Full spectrum	Day 3	79.31	86.96	50.00	86.96
	Day 4	87.10	100	50.00	92.00
	Days 5 to 9	100	100	100	100
RGB	Day 3	79.31	78.26	83.33	85.71
	Day 4	83.87	91.30	62.50	89.36
	Days 5 to 9	96.77	98.00	91.67	98.00

Figure 5 shows example images captured on days 3 and 5 after inoculation corresponding to infected (Figure 5a and 5c) and control (Figure 5b and 5d) olives. The bottom row shows the probabilities given by the model belonging to these olives to each of the classes. These olives have been correctly classified by the model.



a) b) c) d)

Figure 5. Images captured on days 3 (a,b) and 5 (c,d) after inoculation corresponding to infected (a, c) and control (b, d) olives. The bottom row shows the probabilities given by the model of belonging these olives to each class.

3.2. Comparative study

Table 4 shows a comparison of the performance of the proposed model with other alternative models and design options. The best model, taken as a reference for comparison, is the CNN ResNet101 architecture, trained following a two-step training schedule to take advantage of the transfer learning technique. As can be seen from Table 4, the performance of other CNN architectures such as ResNet50 and Densenet121 is inferior, falling even below 80% of a Naïve model in the validation and test sets. This was also the case for all the other architectures tested, except VGG-16, which achieved somewhat better results but still well below the reference model.

Regarding design choices, the exclusion of the TTA post-processing technique produced a worsening of accuracy, showing the effectiveness of the method. An interesting finding is that by using only the RGB bands of the hyperspectral image, the performance decreases significantly, suggesting that the information contained in the other bands is key to the model and demonstrating that the band compression technique was effective at preserving the necessary information. Finally, training all the CNN weights directly in a single step (instead of training in two stages, as explained in Section 2.6) also yielded poor results. In general, several different training techniques (custom training schedule, on-line augmentation, etc.) and specific hyperparameter settings had to be used to achieve good performance even in an imbalanced dataset that is relatively small to be used in a CNN conventionally. As confirmed by the results in Table 3, modifying any of these configurations yielded a significant decrease in performance, proving their importance.

Table 4: Accuracy and F1 comparison for different models and design choices.

Description of changes		Accuracy (%)			F1 (%)
		Train	Val.	Test	Test
Best model	ResNet101	92.67	92.62	91.80	94.90
Other models	ResNet50	85.42	77.05	78.69	88.07
	VGG16	92.71	86.89	87.70	92.15

Other design choices	Densenet121	84.11	78.96	78.69	88.07
	No TTA	93.17	90.98	90.98	94.42
	RGB bands only	90.85	91.67	89.34	93.12
	One-step training	78.69	78.96	78.69	88.07

Several works have been previously published aimed at detecting different types of bruises or diseases in olives using image processing. Most of them use standard computer vision systems based on colour information to discriminate among olives with different quality. Sola-Guirado et al., (2020) determined a bruising index in olives among other colour and geometric parameters. Riquelme et al., (2008) classified olives with eight types of defects. Previously, Diaz et al., (2003; 2004) classified olives in four qualities depending on external defects in real-time. In all cases, good results were achieved, but all related to visible damages and using standard image processing techniques. No previous works have been found related to the detection of anthracnose or any other disease in olive fruit using hyperspectral imaging or deep learning, thus allowing to detect incipient infections or damages before they cause visual symptoms. Therefore, this work is a step forward in the control of fungi diseases in olives after harvest. The sensitivity, that is, the ability of the model to correctly identify infected olives, is high even from the third day after inoculation. From a commercial point of view, these results are relatively good since the objective is to detect the maximum number of infected fruits. It is more tolerable to identify a healthy olive as infected than to allow an olive with such a severe disease as anthracnose to enter the production chain, as it could spread the disease to other olives during storage or transport.

4. Conclusions

Nowadays, machine learning, applied in synergy with a convolutional neural network, is rapidly growing out of necessity to manage an enormous amount of information generates from computer vision techniques. The objective of this work was the early detection of artificially inoculated anthracnose using hyperspectral images through the application of advanced modelling techniques based on Deep Learning (DL) and convolutional neural networks (CNN). ResNet101 architecture was chosen, but it had to be adapted to work with the large number of input bands captured by the hyperspectral system. Images of olives were acquired in the range 450 to 1050 nm. The relatively low number of samples was increased by using data augmentation techniques which resulted in good results of accuracy (91.8%), sensitivity (96.88%), and lower but still reasonable specificity (73.08%) due to false negatives in the first days after the inoculation. From day 5 after inoculation, all accuracy, sensitivity, and specificity achieved 100%, when the damages on the skin of the olives were still minor and had a visual aspect of a small stain. The proposed analysis was successfully tested. If applied in postharvest operations, it could guarantee a reasonable and accurate control of olive anthracnose to safeguard the quality and commercial value of olive and olive oil.

Acknowledgements

This work has been realised in the mainframe of Doctorate courses in Agricultural, Food and Forestry Sciences - XXXII cycle with administrative headquarters based at Università Mediterranea of Reggio Calabria. This work is co-funded by the projects AEI PID2019-107347RR-C31, PID2019-107347RR-C33, IVIA-GVA 51918 and the European Union through the European Regional Development Fund (ERDF) of the Generalitat Valenciana 2014-2020

References

Almeida, J.V. (1899) La gaffa des olives en Portugal. Bulletin of the Mycological. Society of France. 15, 90–94. http://hdl.handle.net/10400.5/13266

Ashraf, S., Kadery, I., Chowdhury, A.A., Mahbub, T.Z., Rahman, R.M. (2019). Fruit Image Classification Using Convolutional Neural Networks. International Journal of Software Innovation, 7, 51-70. http://dx.doi.org/: 10.4018/IJSI.201910010

Bishop, C.M. (2006). Pattern Recognition and Machine Learning. Springer-Verlag, New York (USA).

Cacciola, S.O., Faedda, R., Sinatra, F., Agosteo, G.E., Schena, L., Frisullo, S., Magnano di San Lio, G. (2012). Olive Anthracnose. Journal of Plant Pathology, 94 (1), 29-44. Invited review. Edizioni ETS Pisa. http://dx.doi.org/10.4454/JPPFA2011001.

Carvalho, M.T., Simões-Lopes, P. and Monteiro da Silva, M.J. (2008). Influence of different olive infection ratesof Colletotrichum acutatum on some important olive oil chemical parameters. Acta Horticulturae, 791, 555-558. https://doi.org/10.17660/ActaHortic.2008.791.85

Chen, Y., An, X., Gao, S., Li., Kang, H. (2021) Deep Learning-Based Vision System Combining Detection and Tracking for Fast On-Line Citrus Sorting. Frontiers in Plant Science, 12, 171. https://doi.org/10.3389/fpls.2021.622062.

Cubero, S., Aleixos, N., Moltó, E., Gómez-Sanchis, J., Blasco, J. (2011). Advances in machine vision applications for automatic inspection and quality evaluation of fruits and vegetables. Food and Bioprocess Technology 4(4), 487-504. https://doi.org/10.1007/s11947-010-0411-8

Díaz, R., Faus, G., Blasco, M., Blasco, J., Moltó, E. (2000). The application of a fast algorithm for the classification of olives by machine vision. Food Research International 33, 305-309. https://doi.org/10.1016/S0963-9969(00)00041-7.

Díaz, R., Gil, L., Serrano, G., Blasco, M., Moltó, E., Blasco, J. (2004). Comparison of three algorithms in the classification of table olives by means of Computer Vision. Journal of Food Engineering 61, 101-107. https://doi.org/10.1016/S0260-8774(03)00191-2.

Eckle, E., Schmidt-Hieber, J. (2019). A comparison of deep networks with ReLU activation function and linear spline-type methods. Neural Networks, 110, 232-242. https://doi.org/10.1016/j.neunet.2018.11.005.

Fan, S., Li, J., Zhang, Y., Tian, X., Wang, Q., He, X., Zhang, C., Huang, W. (2020) On line detection of defective apples using computer vision system combined with deep learning methods. Journal of Food Engineering, 286, 110102. https://doi.org/10.1016/j.jfoodeng.2020.110102

FAOSTAT (2016). Food and Agriculture Organization of the United Nations. Available from: http://www.fao.org/faostat/en/#data/QC. Accessed: April 7th, 2021.

Fukushima, K. (1980). Neocognitron: A self-organising neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics 36, 193–202 https://doi.org/10.1007/BF00344251.

Geladi, P. L. M. (2007). Calibration standards and image calibration. In: Grahn, H. F. & Geladi, P. (eds) Techniques and applications of hyperspectral image analysis, pp 203-220, John Wiley & Sons, Chichester, England.

Gómez-Sanchis, J., Blasco, J., Soria-Olivas, E., Lorente, D., Escandell-Montero, P., Martínez-Martínez, J.M., Martínez-Sober, M., Aleixos, N. (2013). Hyperspectral LCTF-based system for classification of decay in mandarins caused by *Penicillium digitatum* and *Penicillium italicum* using the most relevant bands and non-linear classifiers. Postharvest Biology and Technology, 82, 76-86. https://doi.org/10.1016/j.postharvbio.2013.02.011.

Gómez-Sanchis, J., Lorente, D., Soria-Olivas, E., Aleixos, N., Cubero, S., Blasco, J. (2014). Development of a hyperspectral computer vision system based on two liquid crystal tuneable

filters for fruit inspection. Application to detect citrus fruits decay. Food and Bioprocess Technology, 7, 1047-1056. https://doi.org/10.1007/s11947-013-1158-9.

He, K., Zhang, X., Ren, S., Sun, J. (2016) Deep residual learning for image recognition. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 2016-Decem, pages 770-778. IEEE Computer Society. DOI: 10.1109/CVPR.2016.90.

Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q. (2017). Densely Connected Convolutional Networks. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2017, 2261-2269.

Kang, S.P. & Sabarez H.T. (2009). Simple colour image segmentation of bicolour food products for quality measurement. Journal of Food Engineering, 94 (1), 21-25. https://doi.org/10.1016/j.jfoodeng.2009.02.022.

Konstantinos, P.F. (2018). Deep learning models for plant disease detection and Diagnosis. Computers and Electronics in Agriculture, 145, 311-318. https://doi.org/10.1016/j.compag.2018.01.009.

LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E; Hubbard, W., Jackel, L.D. (1989). Backpropagation applied to handwritten zip code recognition. Neural Computation, 1(4), 541-551. https://doi.org/10.1162/neco.1989.1.4.541.

López-García, F., Andreu, G., Blasco, J., Aleixos, N., Valiente, J.M. (2010). Automatic detection of skin defects in citrus fruits using a multivariate image analysis approach. Computers and Electronics in Agriculture, 71, 189-197. https://doi.org/10.1016/j.compag.2010.02.001.

Lorente, D., Aleixos, N., Gómez-Sanchis, J., Cubero, S., García-Navarrete, O.L., Blasco, J. (2012). Recent advances and applications of hyperspectral imaging for fruit and vegetable quality assessment. Food and Bioprocess Technology 5(4), 1121-1142. https://doi.org/10.1007/s11947-011-0725-1.

Lorente, D., Escandell-Montero, P., Cubero, S., Gómez-Sanchis, J., Blasco, J. (2015) Visible-NIR reflectance spectroscopy and manifold learning methods applied to the detection of fungal infections on citrus fruit. Journal of Food Engineering, 163, 17-21. https://doi.org/10.1016/j.jfoodeng.2015.04.010.

Lu, J., Ehsani, R., Shi, Y., Abdulridha, J., de Castro, A.I., Xu, Y. (2017). Field detection of anthracnose crown rot in strawberry using spectroscopy technology. Computers and Electronics in Agriculture, 135, 289–299. https://doi.org/10.1016/j.compag.2017.01.017.

McCulloch, W.S., Pitts, W. A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics 5, 115–133 (1943). https://doi.org/10.1007/BF02478259.

Mohanty, S.P., Hughes, D.P., Salathé, M. (2016). Using Deep Learning for Image-Based Plant Disease Detection. Frontiers in Plant Science, 7(1419). https://doi.org/10.3389/fpls.2016.01419.

Moral, J. and Trapero, A. (2009). Assessing the susceptibility of olive cultivars to anthracnose caused by *Colletotrichum acutatum*. Plant Disease 93, 1028–1036. https://doi.org/10.1094/PDIS-93-10-1028.

Moral, J., Xaviér, C., Roca, L.F., Romero, J., Moreda, W., Trapero A. (2014) La Antracnosis del olivo y su efecto en la calidad del aceite. Grasas y Aceites, 65, e028. https://doi.org/10.3989/gya.110913.

Munera, S., Gómez-Sanchís, J., Aleixos, N., Vila-Francés, J., Colelli, G., Cubero, S., Soler, E., Blasco, J. (2021) Discrimination of common defects in loquat fruit cv. 'Algerie' using hyperspectral imaging and machine learning techniques. Postharvest Biology and Technology, 171, 111356. https://doi.org/10.1016/j.postharvbio.2020.111356.

Nasiri, A., Taheri-Garavand, A., Zhang, Y.D. (2019) Image-based deep learning automated sorting of date fruit. Postharvest Biology and Technology. Postharvest Biology and Technology, 153, 133-141. https://doi.org/10.1016/j.postharvbio.2019.04.003.

Pangallo, S., Nicosia Li Destri, M.G., Agosteo, G.E., Abdelfattah, A., Romeo, F.V., Cacciola, S.O., Rapisarda, P., Schena, L. (2017). Evaluation of a pomegranate peel extract as an alternative means to control olive anthracnose. Phytopathology, 107(12), 1462-1467. http://dx.doi.org/10.1094/PHYTO-04-17-0133-R.

Riquelme, M.T., Barreiro, P., Ruiz-Altisent, M., Valero, C. (2008). Olive classification according to external damage using image analysis. Journal of Food Engineering, 87, 371-379. https://doi.org/10.1016/j.jfoodeng.2007.12.018

Ruder, S. (2016). An overview of gradient descent optimisation algorithms. Arxiv:1609.04747

Schena, L., Mosca, S., Cacciola, S., Faedda, R., Sanzani, S. (2014). Species of the *Colletotrichum gloeosporioides* and *C. Boninense* complexes associated with olive anthracnose. Plant Pathology, 63, 437-446. https://doi.org/10.1111/ppa.12110.

Simonyan K., Zisserman, A. (2015) Very Deep Convolutional Networks for Large-Scale Image Recognition. Computing Research Repository (CoRR), abs/1409.1556. https://arxiv.org/abs/1409.1556. Accessed: April 7th, 2021.

Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., Stefanovic, D. (2016). Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. Computational Intelligence and Neuroscience, Article ID 3289801. https://doi.org/10.1155/2016/328980.

Sola-Guirado, R., Bayano-Tejero, S., Aragón-Rodríguez, F., Bernardi, B., Benalia, S., Castro-García, S. (2020). A smart system for the automatic evaluation of green olives visual quality in the field. Computers and Electronics in Agriculture, 179, 105858. https://doi.org/10.1016/j.compag.2020.105858

Steinbrener, J., Posch, K., Leitner, R. (2019). Hyperspectral fruit and vegetable classification using convolutional neural networks. Computers and Electronics in Agriculture, 162, 364-372. https://doi.org/10.1016/j.compag.2019.04.019.

Sun, Y., Gu, X., Wang, Z., Huang, Y., Wei, Y., Zhang, M., Tu, K., Pan, L. (2015). Growth Simulation and Discrimination of *Botrytis cinerea*, *Rhizopus stolonifer* and *Colletotrichum acutatum* Using Hyperspectral Reflectance Imaging. PLoS ONE 10(12): e0143400. https://doi.org/10.1371/journal.pone.0143400.

Talhinhas, P., Sreenivasaprasad, S., Neves-Martins, J., Oliveira, H. (2005). Molecular and phenotypic analyses reveal association of diverse *Colletotrichum acutatum* groups and a low level of *C. gloeosporioides* with olive anthracnose. Applied and environmental microbiology, 71(6), 2987–2998. https://doi.org/10.1128/AEM.71.6.2987-2998.2005.

Toda, Y., Okura, F. (2019). How Convolutional Neural Networks Diagnose Plant Disease. Plant Phenomics, Article ID 9237136. https://doi.org/10.34133/2019/9237136.

Trapero Casas, A. (2003). La Antracnosis o Aceituna jabonosa en el cultivo del olivo. Vida Rural 168, 46-50.

Wallelign, S., Polceanu, M, and Buche, C. (2018). Soybean Plant Disease Identification Using Convolutional Neural Network. The Thirty-First International Florida Artificial Intelligence Research Society Conference (FLAIRS-31), 146- 151.

Wang, G., Li, W., Aertsen, M., Deprest, J., Ourselin, S., Vercauteren, T. (2018). Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks. Neurocomputing, 338, 34-45, DOI: 10.1016/j.neucom.2019.01.103.

Xiao, J.R., Chung, P.C, Wu, H.Y., Phan Q.H., Yeh, J.A., M.T. Hou (2021) Detection of Strawberry Diseases Using a Convolutional Neural Network. Plants 2021, 10(1), 31. https://doi.org/10.3390/plants10010031.

Yeh, Y., Chung, W., Liao, J., Chung, C., Kuo, Y., Lin, T. (2016). Strawberry foliar anthracnose assessment by hyperspectral imaging. Computers and Electronics in Agriculture, 122, 1–9. https://doi.org/10.1016/j.compag.2016.01.012.

Yuan, L., Yan, P., Han, W., Huang, Y., Wang, B., Zhang, J., Zhang, H., Bao, Z. (2019). Detection of anthracnose in tea plants based on hyperspectral imaging. Computers and Electronics in Agriculture, 167, 105039. https://doi.org/10.1016/j.compag.2019.105039.