Array 18 (2023) 100286

Contents lists available at ScienceDirect

Array

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

On field disease detection in olive tree with vision systems

Pedro Bocca *, Adrian Orellana [a], Carlos Soria [a], Ricardo Carelli [a]

[a] Instituto de Automática, UNSJ-CONICET, San Juan, 5400, Argentina

ARTICLE INFO

Keywords:
Agriculture
Image processing
Diseases
Olive

ABSTRACT

In the present work the capability of convolutional neural networks to extract samples of leaves in images of tree's canopy and detect the presence of different diseases and pests that manifest in deformation, discoloration or direct presence in the leaves, is studied. The sample obtained along with its location and sampling date, allows a mapping of the diseases in the field. This mapping capability will allow better decisions to be made when fighting these canopy diseases. An example of those are fungus and Aceria oleae in olive leaves. The study begins with the analysis of a data set generated in the laboratory and divided into healthy and faulty parts. The images were captured with a RGB and a multi-spectral with the blue, green, red, near infrared and red border spectra. They were taken in an image laboratory with a white background and led lighting. The objective was to carry out tests to determine the impact of each spectral channel and the possibility of using different types of cameras for the detection of diseases, as well as important factors to consider for its application in the field. Then, Mask rcnn R 50 FPN 3 was used to obtain segmented leaves and Fast-r cnn inception v2 to detect leaves. Then the detected or segmented leaves were classified with the Inception V3 network to determine which were healthy and which were diseased. With, the combination of these tools, it is possible to determine the disease level of an olive tree in the field.

1. Introduction

Diseases in fruit trees represent a pervasive problem that considerably reduce the productive capacity of them [1,2]. Also, the abuse of pesticide to control the diseases represents a significant environmental impact and economic cost [3–5]. The estimation of the disease level in fruit trees is currently made by direct observation by an expert. This procedure is time-consuming and difficult to be applied in intensive agriculture, which motivates the development of tools for automatic evaluation of the amount of pesticides required to be applied locally on the tree's crop. The proper disease monitoring allows early detection, propagation control and analysis of their evolution based on the processing of data that can be automatically collected. This allows for better decision-making when applying pesticide agents. The use of detection systems by means of artificial intelligence and deep learning, is nowadays present in different aspects of daily life and in the professional field [6,7]. The use of image processing for detection and classification of diseases in agriculture has a great impact on early diagnosis and control [8–12]. The performance of convolutional neural networks to classify images has improved over time [13], and have shown a good performance in the detection of diseases on leaves [14, 15], as well as the impact of different spectra on its detection [16–18].

In [9] a method for the detection of diseases in RGB pictures of leaves is proposed. It uses K-means for the extraction of diseases features and an artificial neural network (ANNs) for its classification.

In [16] the potential of hyperspectral sensor system in the detection of fungal leaf diseases is studied for sugar beet. The detection of disease-specific spectral signatures is desired.

In [19] the classification of Neofabraea fungus and Spilocaea oleaginea fungus in a sample of olive leaf through the extraction of textures features is analyzed. In the presented work, the detection of diseases in an uncontrolled environment with a wide variety of light conditions and leaf orientation is analyzed.

In [20–22] different research are carried out to detect diseases through artificial intelligence and pattern recognition. In all cases, the samples are analyzed in controlled environments.

In [23] convolutional neural networks are analyzed to detect different diseases in olive trees. As in other studies, a manually prepared high-quality dataset is used to assess the performance of convolutional neural networks to classify diseases.

In this study, it is analyzed the impact of different individual spectrum for the training of convolutional neural networks. In addition, the effect of segmentation in the training of convolutional neural networks is also studied. Then, different structures of these networks are

* Corresponding author.
E-mail address: pbocca@inaut.unsj.edu.ar (P. Bocca).

https://doi.org/10.1016/j.array.2023.100286
Received 6 December 2022; Received in revised form 16 February 2023; Accepted 26 February 2023
Available online 1 March 2023
2590-0056/© 2023 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

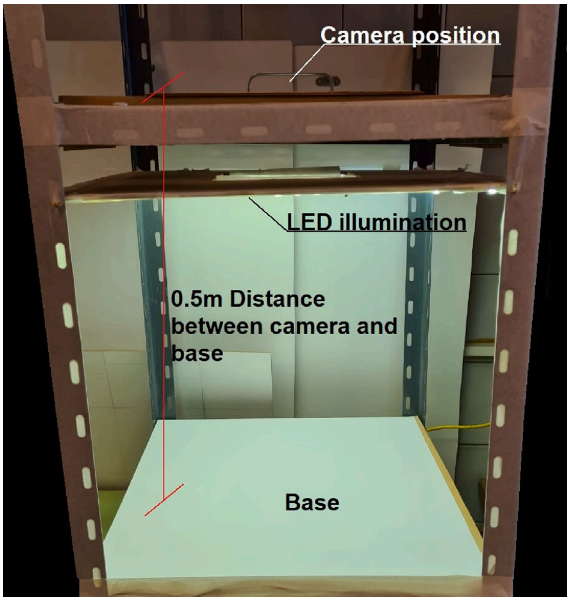
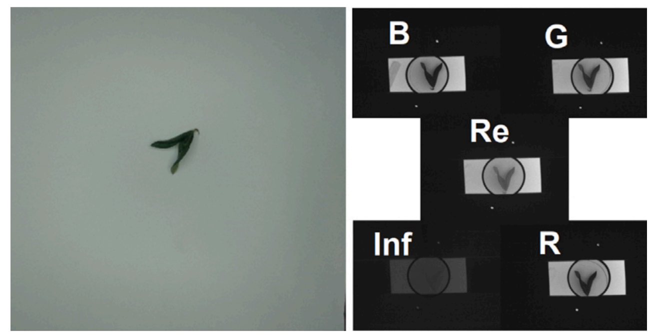


Fig. 1. Image laboratory setup.

(a) Sony DSC-W830 camera (b) Micasense Rededge 3 camera

Fig. 2. Cameras used to create the dataset.



(a) RGB camera image (b) multi-spectral camera image

Fig. 3. Pictures obtained by both cameras.

(a) RGB camera (b) multi-spectral camera

Fig. 4. Pictures obtained by both cameras after preprocessing.

considered for the combination of different spectra with stereo effect, improving the capacity of the neural network to work with multi-image systems. Finally, the evaluation of the disease level of an entire tree is analyzed through the detection or intrinsic segmentation of its leaves. As the images come from real environment, those detections have a variety of illuminations, orientations and pixel range, which produce false detections like shadows, branches, fruits that are outside the classifier. The system must work with all these patterns and eliminate or ignore those that do not have to be analyzed and thus obtain the disease level of the tree in the field to apply the appropriate treatment. All the necessary processes to evaluate the disease of a tree in the field are carried out by the presented research, which includes the acquisition of images, their preprocessing, the selection of objects to be analyzed, the reduction of dimensions and finally the classification. The aforementioned studies and results allow the creation of a tool for field use capable of evaluating the health condition of olive trees and store the information with its time-location data, to obtain an adequate analysis of the diseases, their concentration and evolution.

2. Obtaining and preparing the dataset

To build the data base, the structure shown in Fig. 1 was used with an RGB camera or a multi-spectral camera, mounted on top. The olive leaves samples were taken at INTA-San Juan (National Institute of Agricultural Technology in Argentina, https://inta.gob.ar/sanjuan) and classified by professionals in the area of diseases in olive trees.

In this article, the influence of different spectra was analyzed. To achieve this, a multi spectral camera with 16-bit 1280 × 960 pixels images was used as well as a standard RGB camera with 24-bit 5152 × 3864 pixels images. Both of them are shown in Fig. 2.

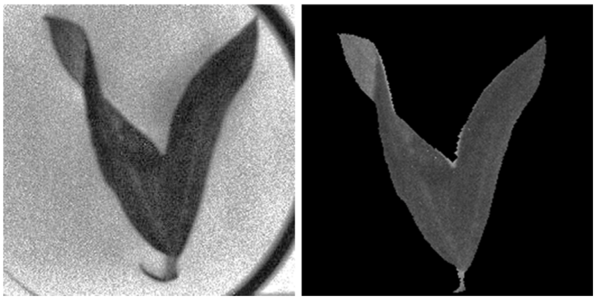
Fig. 3(a) shows an image obtained with the RGB camera and the multi-spectral camera. Multi spectral camera has the Red (R), Green (G) and Blue (B) spectra as well and includes Infra-red (Inf) and Red edge (Re) spectra. All the spectra can be seen on Fig. 3(b).

2.1. Preprocessing

Before feeding the neural network with the captured images, a preprocessing was performed to increase the portion of image occupied by the leaves and the relevant pixels for obtaining the features. In the case of the multi spectral camera, the stereo effect was compensated so that the leaves were approximately in the same position in the 5 images, this work was made with OpenCV. The result of this process is shown in Fig. 4(a) for the RGB camera and in Fig. 4(b) for the blue spectra of the multi-spectral camera. In both cases, an image of 260 × 260 pixels was obtained and then reduced to 150 × 150 pixels before being fed to the neural network. With this, a dataset of 500 images was generated with each camera and divided into healthy and faulty parts. Then it was divided into training with an 80% and validation with 20%. Data augmentation that consists on small modifications was used to artificially increase the dataset, that was zoom in and out, rotation, mirror in different values to increase the dataset to 1000 samples. To test the consistency of the results, the dataset was shuffled, and randomized training and validation sets were assembled and tested. All these sets produced similar results.

2.2. Segmentation of leaves

The leaves were segmented and separated with a black background to measure the impact of this process when training neural networks.



(a) Image before segmenting the leaves. (b) Image after segmenting the leaves.

Fig. 5. Images of the leaves before and after segmenting.

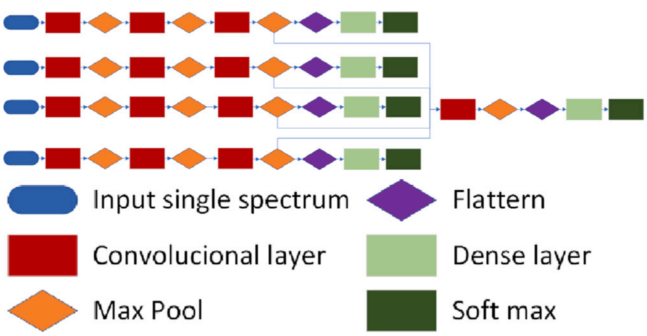


Fig. 6. First neural network structure analyzed.

In Figs. 5(a) and 5(b) the result of this process for one spectrum can be seen.

3. Neural network structure

The three main neural network structures that were tested are shown below. As it is seen in Fig. 4(b) the illumination used to generate the dataset has a low presence of infra-red spectra, for that reason this spectra was ignored in the following tests. Fig. 6 shows the first structure used for images from the multi-spectral camera. It has 4 gray-scale image entries corresponding to each spectrum channel and 5 outputs, 4 of which analyze each spectrum individually and the last analyses a combination of all of them. The combination of the four spectra is performed after the second max pooling process.

The second structure used for the multi-spectral camera is shown in Fig. 7. In this case the neural network is similar to the first one. It has 4 gray-scale image entries corresponding to each spectrum channel and 5 outputs, 4 of which analyze each spectrum individually, and the last analyses a combination of all of them. The difference with the first neural network is that the combination of the four spectra is made on first dense layer instead of being combined in the second process of max pooling.

Finally, Fig. 8 shows the third neural network used to be trained with the images that contains all spectra combined. In this case, a multispectral image with 3 or 4 spectra (RGB or RGBRe[1]) is used and it has 1 output with the result.

4. Results and analysis

In this part of the work, it was analyzed the detection of fungus, Saissetia oleae, Spilocaea oleagin, Aceria oleae and toxicity. The dataset

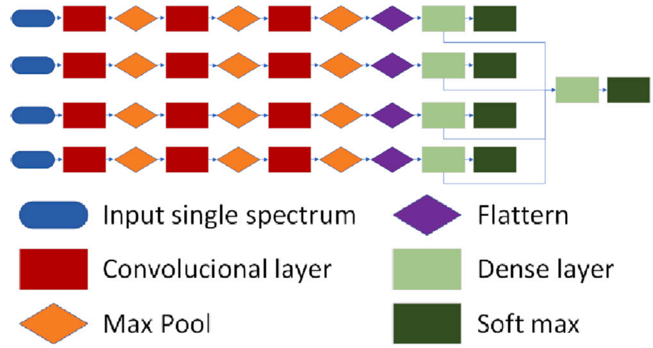


Fig. 7. Second neural network structure analyzed.

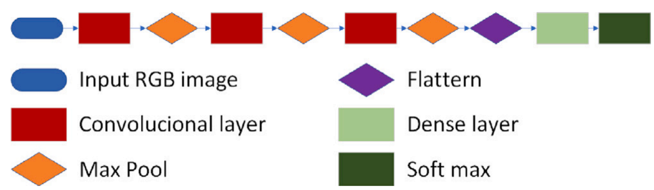


Fig. 8. Third neural network structure analyzed.

obtained from the multi-spectral camera was analyzed through the first neural network shown in Fig. 6, providing the following results. The training result of channel blue is given in Fig. 9(a), where it can be seen that the training reaches 80% accuracy, but the validation set is about 72%. This indicates an over training of 8 percent points and that the real accuracy is about 72%. Besides that, increasing the training cycles will not improve the accuracy of the system.

In Fig. 9(b), corresponding to the results of training the system with the channel green, it can be seen that the training reaches 85% accuracy, but the validation set is of 78%. As with channel blue, it is observed a significant over training, and a real accuracy approximately of 78%.

With channel red observed in Fig. 10(a) we obtain similar results with channels green and blue. Finally, in channel red edge shown in Fig. 10(b) it can be seen that the training reaches 90% accuracy, but the validation set is about 83%. It is concluded that this channel is the one that provided the best results by itself, and therefore it is considered to contain the most significant information of the faulty parts.

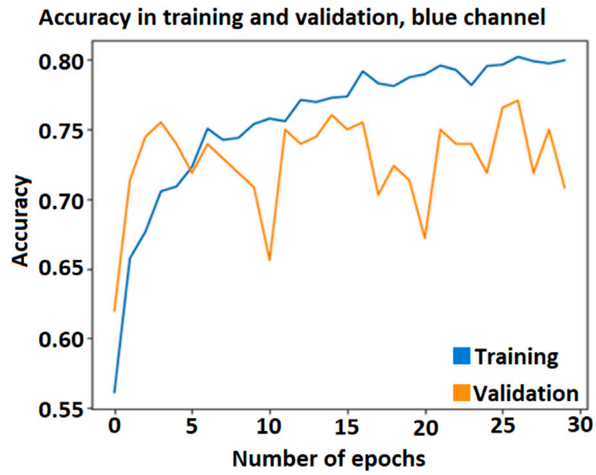
In Fig. 11(a), corresponding to the results of the system training with the previous 4 channels combined, it can be observed that the training reaches 90% accuracy, but the validation set is of 85%. In this case, the over training is reduced and reaches the best percentage in validation. In addition, if it is compared with the results obtained by the RGB camera shown in Fig. 11(b), it can be seen that the results are similar, thus being the multi-spectral again susceptible to over training. In this case a larger data base may help for getting a correct training.

4.1. Impact of segmentation in training

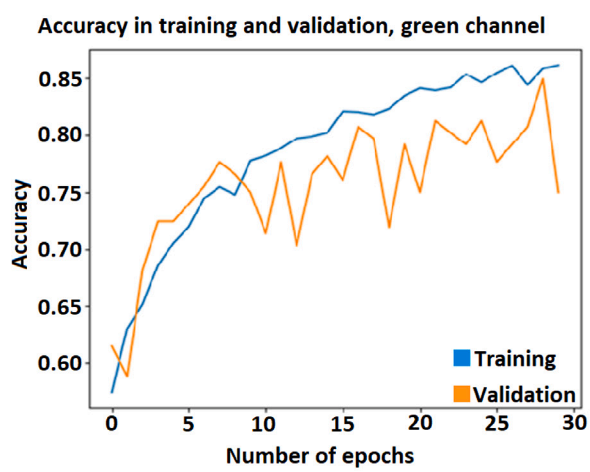
The dataset with the segmentation was used to train the first neural network of Fig. 6. The results for channels green and blue are observed in Figs. 12(a) and 12(b). Unlike the previous training, it can be seen that it does not suffer from over training, and the accuracy of the system stabilizes at approximately 76%, obtaining a result similar to the validation of the system without segmentation. For that reason, the impact of channels green and blue for the detection of faulty parts can be considered to be low.

In Fig. 13(a) it is shown the accuracy corresponding to the red channel. As in the previous case, almost no over training is observed and the accuracy is about 85%, getting an improvement of 11% in

[1] Red, green, blue and red edge combined

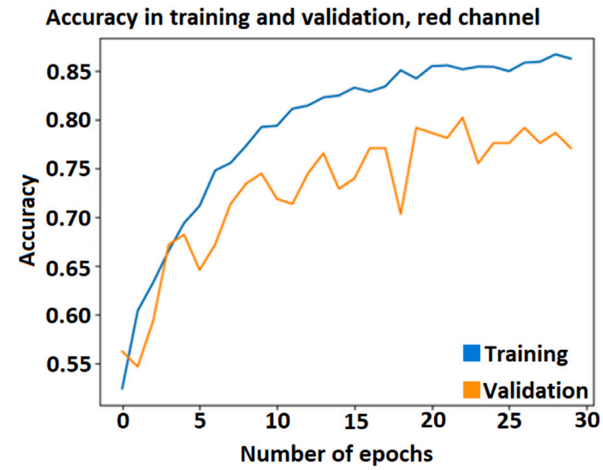


(a) Channel blue training

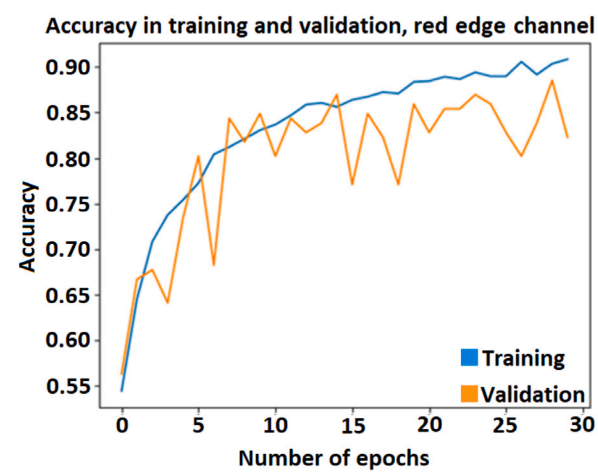


(b) Channel green training

Fig. 9. Training and validation results. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



(a) Channel red training



(b) Channel red edge training

Fig. 10. Training and validation results. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

comparison with validation without segmentation. Fig. 13(b) shows the case for red edge channel. The over training reduction happens as well as in the other channels, and it is obtained an accuracy of 82%. With segmentation red channel gets a significant improvement and all channels exhibit the overtraining problem. Also, red and red edge channels are the best ones to detect faulty parts and both channels have space for further improvement, so that they can continue the training to increase the accuracy of the detection.

Fig. 14 shows the result for the combination of the four channels. It can be observed that it remains above the Red channel with 86% of accuracy and similar to the individual channels, it reduces the overtraining and seems to have space for further improvement. Therefore, it is concluded that, by eliminating the environment in the leave by means of the segmentation process, the network is forced to analyze the internal elements, thus improving the extraction of characteristics related to faulty parts.

4.2. Second proposed architecture for the multi-spectral camera

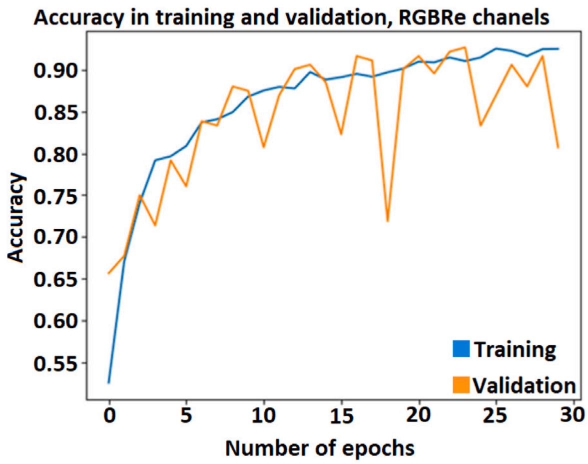
The training of the structure shown in Fig. 7 was tested to analyze the impact on the results by changing the position in which the channels are combined. The result for the combination of the four channels on the new structure can be seen on Fig. 15.

When training the neural network, it is observed that the behavior of the individual channels red, green, blue and red edge is the same as in the case of the neural network of Fig. 6, as expected, which can be seen in Figs. 16(a) and 16(b). However, as seen in Fig. 16(b), the result of the combination of the four channels shows a significant improvement with respect to the Red channel, obtaining an accuracy of 88%. Then, it is concluded that making decisions from the partial decisions of each channel has a better effect than taking them from the characteristics of each channel. This in turn, improves the independence of each channel reducing the stereo effects of the camera.

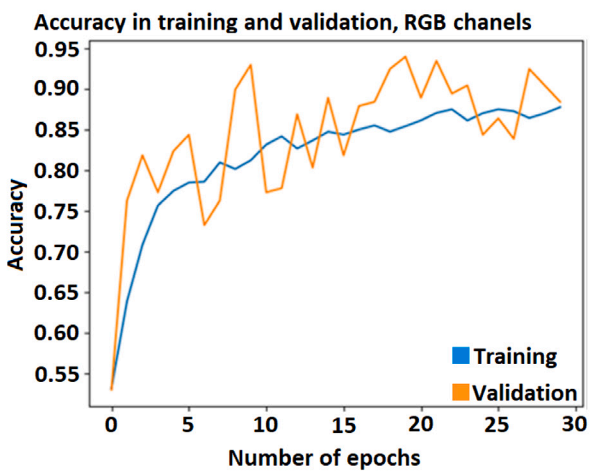
4.3. Recomposition of images

Taking advantage of the segmentation of the leaves, the images from each of the spectra were combined to form new RGB and RGBBRe multi-channel images. Then, they were tested on the third neural network of Fig. 8. The result of RGB combination is observed in Fig. 17.

Figs. 18 and 19 show the result of the training for the RGB and RGBRe[1] image sets from the multi-spectral camera. It can be seen that the RGBRe[1] system has 2% more accuracy than the RGB system, reaching up to 86% of accuracy, in addition to having a better performance in the validation set. It should be noted that when the tests were done with the non-segmented images, the over training was very

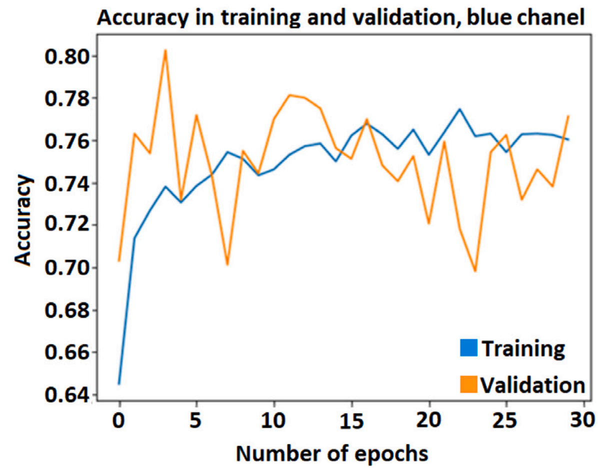


(a) Multi-spectral camera

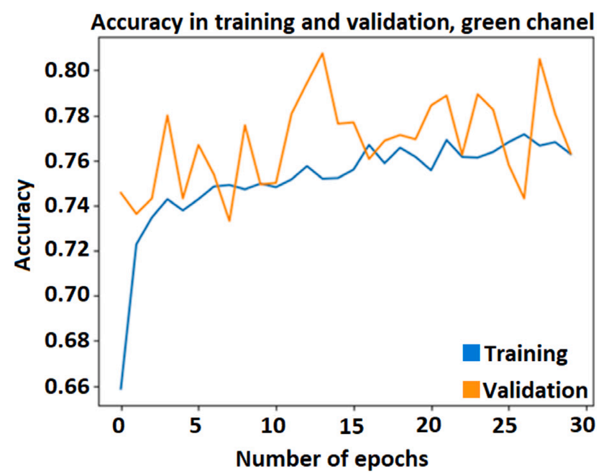


(b) RGB camera

Fig. 11. Training and validation results. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



(a) Channel blue training



(b) Channel green training

Fig. 12. Training and validation results. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Result of training in the different networks with segmentation.

Chanel	Net 1 Fig. 6	Net 2 Fig. 7	Net 3 RGB camera Fig. 8	Net 3 Multiexpectral camera combination Fig. 8
Red	85,00%	84,90%	–	–
Green	76,10%	74,90%	–	–
Blue	75,80%	75,60%	–	–
Red edge	83,30%	83,90%	–	–
RGB	–	–	87,20%	87,80%
RGBRe[1]	86,20%	88,40%	–	89,90%

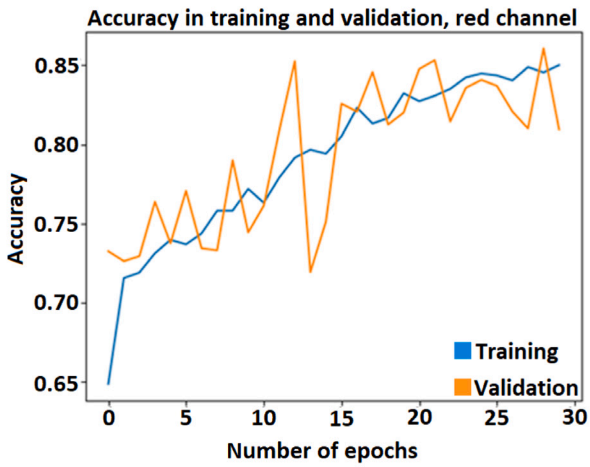
high, so these tests were discarded. However, the segmentation allows to combine all the channels and feed the neural network with a single multichannel image.

Finally, Table 1 gives the results of all the previously discussed trainings.

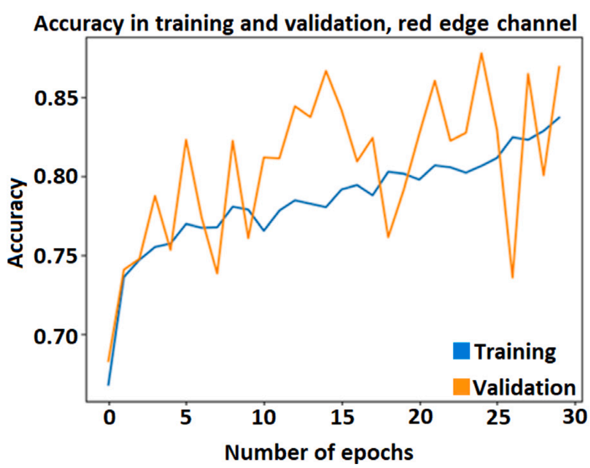
As seen in the bibliography, artificial intelligence is useful in detecting faulty parts in olive leaves. This can be seen in [24] where 97% accuracy was achieved using a combination of the ViT model (Visual Transformer) and the VGG-16 (Visual Geometry Group), or in [25] where 91.8% accuracy was achieved with ResNet101. Finally, in the study [26], it was observed that inception V3 had an accuracy of 98% using the SGD (Stochastic Gradient Descent) as optimizer.

The objective of this study is not to improve the precision of convolutional neural networks to detect faulty parts, but to determine which factors are important to apply them in the field. In this case it was observed that:

- The longer wavelengths, such as red and red edge, highlight faulty parts the most. Sunlight has a good presence of these wavelengths so the use of additional lighting is not necessary. However, in closed environments such as greenhouses or at night, this aspect must be taken into account and adequate lighting must be used to detect faulty parts.
- The segmentation of the samples helps to reduce the overtraining between the training set and the validation set. In case of having an insufficient dataset, this factor can be used together with data augmentation like rotation, flip, zoom, etc.
- At least 150 pixels are required at the longest length of the image to detect faulty parts on the leaves. Considering the size difference between the leaves and the tree, many high-resolution images will be required to properly evaluate the tree.



(a) Channel red training



(b) Channel red edge training

Fig. 13. Training and validation results.

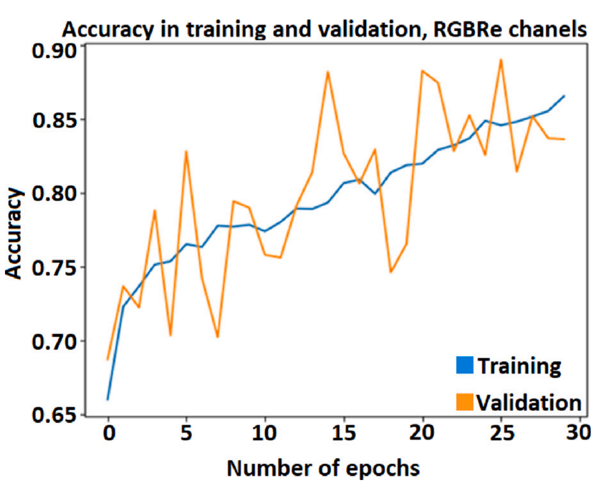


Fig. 14. Training and validation results from RGBRe channels.

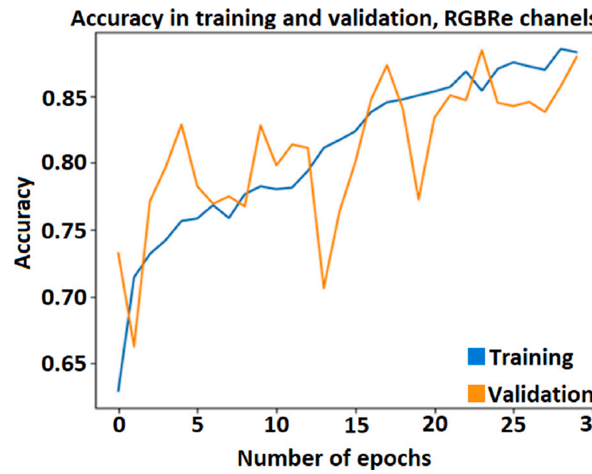
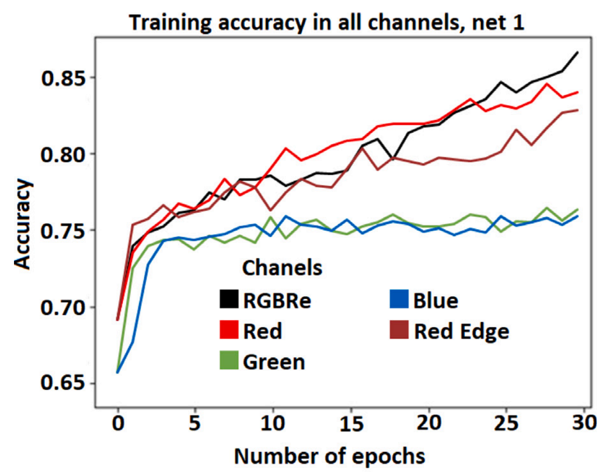
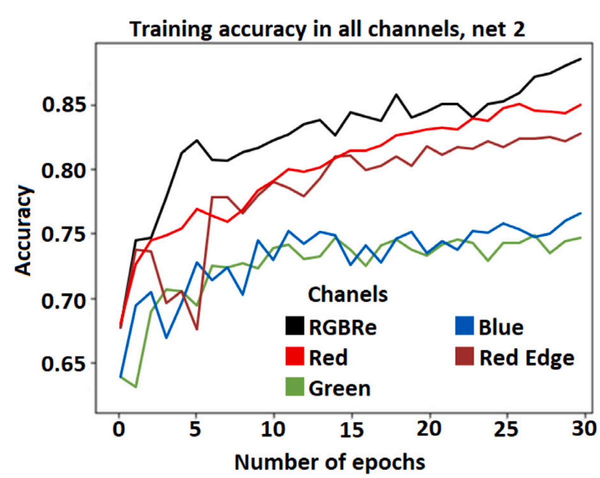


Fig. 15. Training and validation results from RGBRe channels.



(a) Net 1, all channels training



(b) Net 2, all channels training

Fig. 16. Net 1 and 2 comparison. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5. Taking a sample of leaves to evaluate the disease level of the tree

Olive trees have thousands of leaves, thus detecting and evaluating all of them is not practical. To solve that problem, a representative sample of them was used. The objective was to detect a random number of leaves big enough to represent the health level of a part of the tree. The net should detect healthy and faulty parts alike. In this section,

Fig. 17. Combination of red, green and blue channels. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Fig. 20. Group of leaves manually leveled.

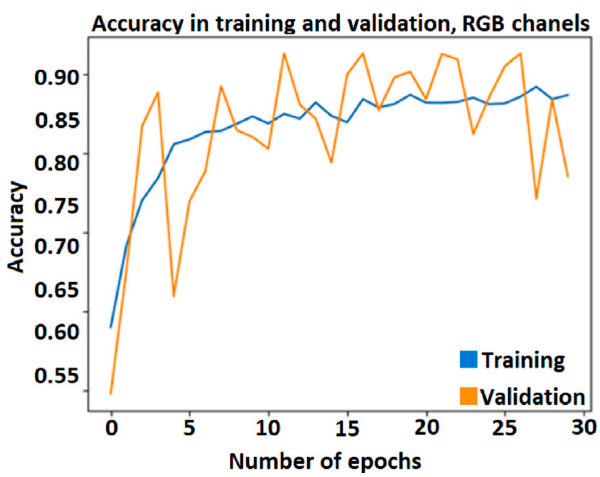


Fig. 18. RGB combination.

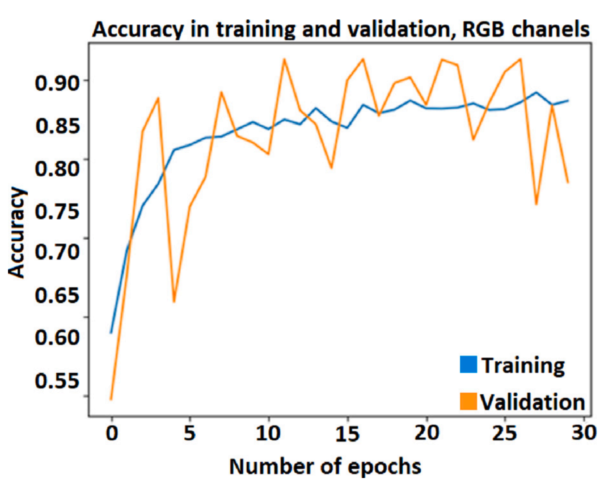


Fig. 19. RGBRe combination.

it is analyzed the capability of neural networks to detect or perform intrinsic segmentation on leaves.

5.1. Main neural network structures

To detect olive leaves it was used the faster-rcnn-inception-v2 neural network [27], and for intrinsic segmentation it was applied the mask-rcnn-R-50-FPN-3x from Detectron 2 [28]. Both networks were retrained with a dataset that contains healthy and sick leaves alike. With this, was expected that both neural networks could generate a representative sample of leaves which could be used to determinate the health level of the tree. Finely, Inception V3 convolutional neural network was used to classify the detected leaves on the different categories as this network has shown good result on leaves disease classifications [29,30].

5.1.1. Fast-r cnn inception v2

Fast-r cnn inception v2 object detection network depends on region proposal algorithms to hypothesize object locations. This network was chosen because it has a good detection of leaves and is compatible with Intel Movidius Neural Compute Stick 2.

The network was retrained with a 150-image with 1600 leaves manually detected as shown in Fig. 20. The dataset was divided into 80% training and 20% validation. The images were taken at different hours along the year with natural light.

5.1.2. Mask rcnn R 50 FPN 3x

Mask rcnn is an object instance segmentation [31]. This neural network extends Faster R-CNN by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition. This framework is capable of detecting and segmenting individual leaves and to eliminate all the unnecessary information around it. This allows the classifier to evaluate only the detected leave and avoid the effect of surrounding elements. The same set of images of the Fast-r cnn were used to train this network with the leaves manually segmented. This can be seen in Fig. 21.

Fig. 21. Group of leaves manually segmented.

(a) Detected healthy leave with sick leave next to it   (b) Segmented healthy leave

Fig. 22. Comparison between detection and segmentation.

(a) Excessive light   (b) Fruit detection

Fig. 23. non-classifiable detections.

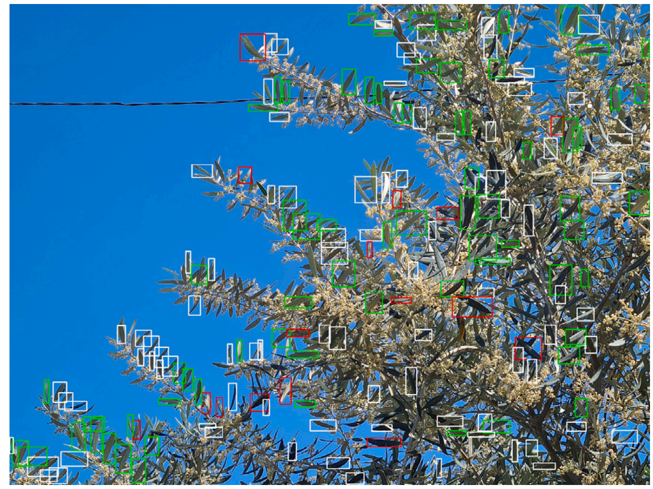


Fig. 24. Part of a tree with a part of the leaves detected and classified. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.1.3. Inception V3

As seen in [26], Inception v3 is more than 90% accurate in detecting faulty parts. That is why this network is used in combination with the others to evaluate the tree. This network will be retrained with a dataset of 3000 leaves divided into healthy, faulty parts and unclassifiable. The training set will be 80% and the validation set 20%. The dataset was increased to obtain 1500 samples per category, reaching 4500 samples with data augmentation. As in the previous cases, the dataset was mixed and recreated to detect inconsistencies in the network training.

5.2. Detection and classification of faulty parts in leaves

In the beginning of this work, it was analyzed the capability of CNN to detect faulty parts on leaves. Then, two neural networks were retrained to extract a sample of leaves on trees. Later, both networks were used to evaluate the disease level of a part of an olive tree crop. In this case the classifier was retrained to detect three classes: healthy, sick and non-classifiable leaves. Both, the Mask rcnn and the Fast-r cnn can perfectly detect a sample of leaves, but through the experiment it was observed that both of them can detect other objects than leaves, like the olive fruits and flowers. The classifier had to recognize those false detections and ignore them. Leaves out of focus were considered non classifiable as well. The classifier was retrained with a new dataset generated by the detector and the intrinsic segmenter.

5.2.1. Difference between detection and intrinsic segmentation

In the first part of the work, it was found that the segmentation of the leaves has a big impact in the neural network training. Using leaves obtained in real conditions and retraining the Inception V3, a bigger dataset with at least 1000 images of each class is required. The retraining result of segmented and detected leaves was similar and no advantage between each other was detected. The only difference was that with only detection, the classifier evaluates the detected leaf and the ones that are close to it, so the presence of disease in the detected leaf or any one close to it was considered as faulty part. With intrinsic segmentation, all surrounding elements are eliminated so only the detected leaf was analyzed. This comparison can be seen in Fig. 22. Therefore, the selection of the neural network has depended on the requirements of the embedded system hardware.

5.2.2. Incorrect detections

Detection and intrinsic segmentation are not perfect, so a specific category is created for detection errors. Those errors can be the detection of flowers or fruits instead of leaves, an inadequate angle for the correct detection of faulty parts, too much or too low amount of light. Sheets that are not in focus are also discarded. A new category with all

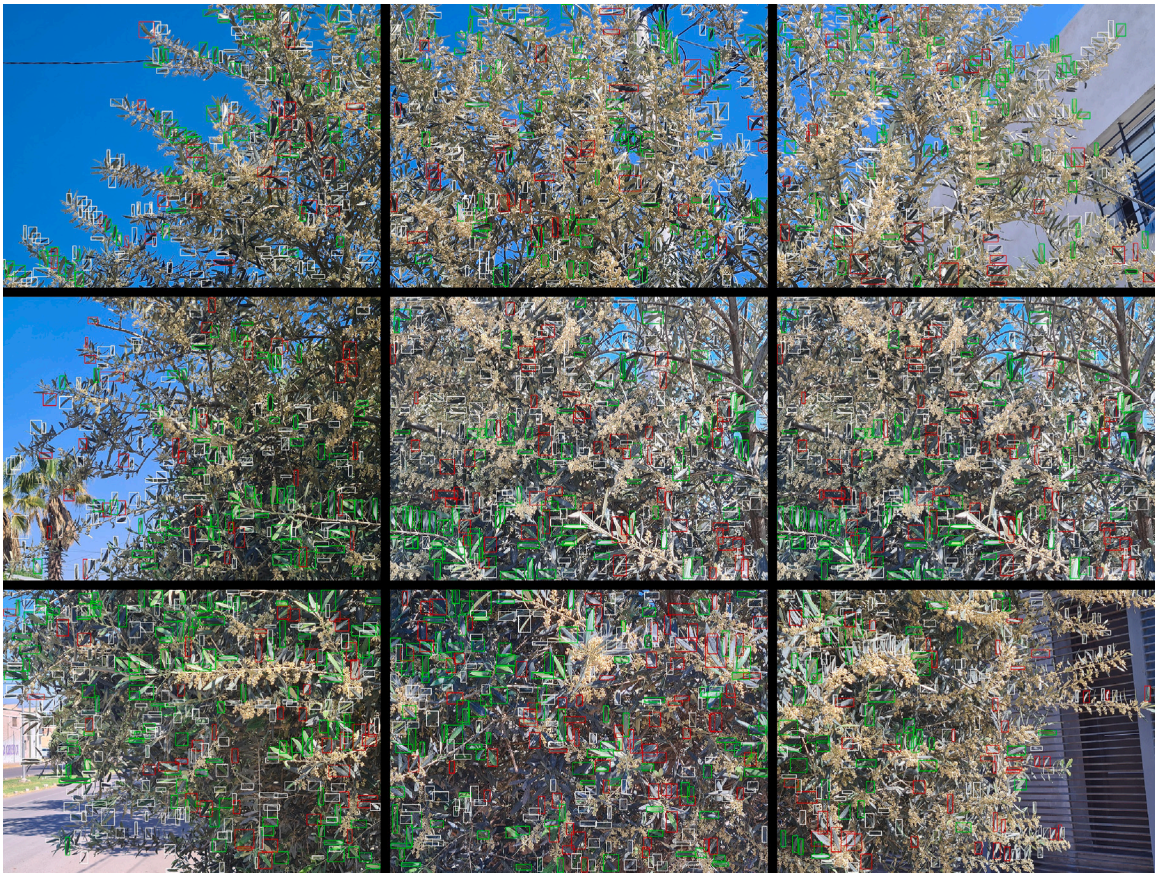


Fig. 25. Olive tree with leaves detected and classified. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

non-classifiable detections is created for the classifier to override. Also, the leaves detected that do not have enough pixels are also discarded. Examples of erroneous detection and excessive light are shown in Fig. 23.

5.2.3. Analyzing the disease level of an olive tree

The main objective of this work was to perform an analysis of an olive tree and detect the presence of disease and the concentration of it in the tree. This is of great importance in precise agriculture, and for developing precise autonomous fumigation systems. To achieve that goal, it was necessary to take a picture or several pictures of the tree that contain enough pixels of the leaves. The classifier requires 150 pixels in the longer axis of the leave to property classify it. It is recommended to have 200 pixels in the longer axis of the leaves and then reduce it to 150 pixels. Due to the strong relation between the size of the tree and the size of the leaves, several images in high resolution were taken and analyzed to make a proper evaluation of the tree. Due to that restriction, it was used an RGB camera with high resolution.

To evaluate a whole tree canopy, several pictures were needed. Then each picture was divided in 9 parts and a sample of leaves were taken and extracted from each image. Then each leaf was processed by the classifier to determine the amount of healthy and faulty parts. The process is shown on Fig. 24. Faulty parts are in red, healthy ones in green and the unclassifiable in white. It is considered non-classifiable the leaves that have not enough pixels to be processed, the ones that are out of focus and the wrong selections.

As the position of each picture is known, the disease level of all the tree and the concentration on each part of it can be determined. This is shown on Fig. 25.

As shown in Figs. 24 and 25, the proper combination of both neural networks can perform the disease assessment of a whole tree. In this case the photos must be taken manually and pass the images through the tool to evaluate it, being able to save the results with their corresponding date and location, which contributes to precision agriculture techniques.

5.3. Embedded system

The main goal of this work was to generate a tool that could evaluate the disease level of a tree in the field, by the analysis of the different parts of it and then as a whole. To do so, it was needed to execute a neural network to extract a representative sample of leaves and a classifier to evaluate the amount of healthy and sick leaves of the sample. Then, export both neural network in a device that could be easily transported and complemented with a gps and a camera module.

Both, Faster-rcnn-inception-v2 and Inception V3 neural networks, were exported to OpenVino and executed on an Intel Movidius Neural Compute Stick 2 mounted on a Raspberry pi 3–4. The hardware used is incompatible with any mask rcnn system so only faster R-CNN with Inception V3 was used in the embedded system. The analysis of each picture takes between 9–13 s of processing. As it was required nine pictures to analyze the entire tree, the work could be done in less than 100 s depending on the number of samples that have been taken. Thanks to the high parallel processing capabilities of the raspberry pi 3–4 and the Neural Compute Stick, the processing time can be reduced by the addition of more Neural Compute Stick. The results were stored on a json file, allowing the mapping and time-analysis of the disease evolution. The Flowchart of the designed software is shown in Fig. 26. A high-quality image with 4032×3024 pixels is obtained via a camera or extracted from a folder. This image is divided into 9 parts and stored in a container. Then, the main program starts 3 threats and each one sends an image to the neural stick and gets the detected leaves. Each

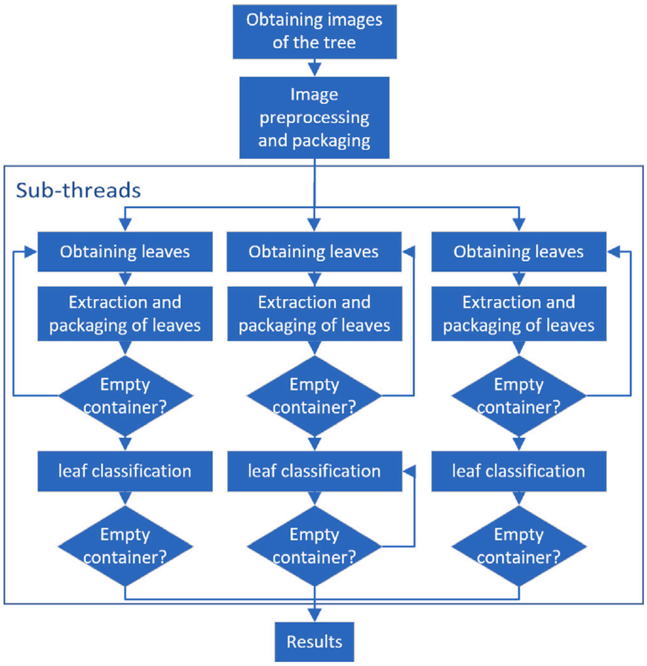


Fig. 26. Software flowchart.

threat extracts the detected leaves, resize them to 150 × 150 pixels, discards the ones that had not enough pixels to be classified and stores them in a new container. When the extraction process is complete, each threat sends a packet of leaves for the neural stick to classify and store the result. When all the leaves are classified the amount of healthy, faulty parts, non classified, discarded leaves is stored and shown on screen.

6. Conclusions

In the first part of the present work, it was presented an analysis of convolutional neural networks and their efficiency in detecting diseases in leaves, that manifest in a visual alteration of the healthy leaves or faulty parts. From the various studies carried out, it was concluded that the convolutional neural network of Fig. 7 with segmented leaves had the better performance reaching up to 88% with low overtraining. Also, it could be observed that the red and red border spectra provide the most significant information on the health of the plant. In addition, the leaves segmentation, implemented to eliminate unwanted information, considerably reduces overtraining. Then with a bigger dataset of in-field samples no difference could be detected between segmented and detected leaves. Finally, when the spectra must be analyzed separately, it is better for the network to analyze and classify each spectrum individually and then, from these partial conclusions, to analyze all the spectra as a whole.

In the second part of the work, the implementation of neural networks to detect and evaluate the disease level of an olive tree was analyzed. It could be observed that both detection and semantic segmentation networks can obtain a representative sample of leaves that can be classified determining the disease level of a part of the tree. Then, the system to evaluate the health status of an olive tree was exported to an embedded system that could be used in on-field applications by an operator.

7. Future work

In future work a dataset with the main diseases will be made to evaluate the presence of the different diseases on the leaves of the olive trees and determine the proper action to be taken. Also, an improvement in the detection or intrinsic segmentation to detect the presence of some disease that is present in other parts of the tree will be studied. Additionally, the device will be mounted on an autonomous quadricycle [32] to avoid the use of an operator and to improve the disease mapping in extensive fields. Finally, the capability to control a whole autonomous fumigation system [33] with the information provided with this detection system will be developed and analyzed.

CRediT authorship contribution statement

Pedro Bocca: Conceptualization, Methodology, Software, Investigation, Data curation, Writing – original draft, Visualization. **Adrian Orellana:** Software, Validation, Formal analysis, Writing – review & editing, Supervision. **Carlos Soria:** Resources, Writing – review & editing, Supervision. **Ricardo Carelli:** Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Ricardo Carelli reports financial support was provided by Consejo Nacional de Investgaciones Científcas y Técnicas.

Data availability

Data will be made available on request.

Acknowledgment

The work has been partially funded by CONICET-Argentina (Consejo Nacional de Investigaciones Científicas *y* Técnicas).

References

[1] Haniotakis GE, et al. Olive pest control: present status and prospects. IOBC Wprs Bull 2005;28(9):1.
[2] Morelli M, García-Madero JM, Jos Á, Saldarelli P, Dongiovanni C, Kovacova M, Saponari M, Baños Arjona A, Hackl E, Webb S, et al. Xylella fastidiosa in olive: A review of control attempts and current management. Microorganisms 2021;9(8):1771.
[3] Pimentel D, Burgess M. Environmental and economic costs of the application of pesticides primarily in the United States. In: Integrated pest management. Springer; 2014, p. 47–71.
[4] Sarkar S, Gil JDB, Keeley J, Jansen K. The use of pesticides in developing countries and their impact on health and the right to food. European Union; 2021.
[5] Sharma A, Kumar V, Shahzad B, Tanveer M, Sidhu GPS, Handa N, Kohli SK, Yadav P, Bali AS, Parihar RD, et al. Worldwide pesticide usage and its impacts on ecosystem. SN Appl Sci 2019;1:1–16.
[6] Aggarwal K, Mijwil MM, Al-Mistarehi A-H, Alomari S, Gök M, Alaabdin AMZ, Abdulrhman SH, et al. Has the future started? The current growth of artificial intelligence, machine learning, and deep learning. Iraqi J Comput Sci Math 2022;3(1):115–23.
[7] Vaid S, Kalantar R, Bhandari M. Deep learning COVID-19 detection bias: accuracy through artificial intelligence. Int Orthop 2020;44:1539–42.
[8] Al Bashish D, Braik M, Bani-Ahmad S. A framework for detection and classification of plant leaf and stem diseases. In: 2010 International conference on signal and image processing. IEEE; 2010, p. 113–8.
[9] Al-Hiary H, Bani-Ahmad S, Reyalat M, Braik M, ALRahamneh Z. Fast and accurate detection and classification of plant diseases. Int J Comput Appl 2011;17(1):31–8.
[10] Gulhane VA, Gurjar AA. Detection of diseases on cotton leaves and its possible diagnosis. Int J Image Process (IJIP) 2011;5(5):590–8.
[11] Kartikeyan P, Shrivastava G. Review on emerging trends in detection of plant diseases using image processing with machine learning. Int J Comput Appl 2021;975(8887).
[12] Li L, Zhang S, Wang B. Plant disease detection and classification by deep learning—a review. IEEE Access 2021;9:56683–98.
[13] Giunchiglia E, Lukasiewicz T. Multi-label classification neural networks with hard logical constraints. J Artificial Intelligence Res 2021;72:759–818.

[14] Berger J, Preussler C, Agostini JP. Identificación de síntomas de Huanglongbing en hojas de cítricos mediante técnicas de deep learning. Electr J SADIO 2019;18.

[15] Lu J, Tan L, Jiang H. Review on convolutional neural network (CNN) applied to plant leaf disease classification. Agriculture 2021;11(8):707.

[16] Mahlein A-K, Steiner U, Dehne H-W, Oerke E-C. Spectral signatures of sugar beet leaves for the detection and differentiation of diseases. Precis Agric 2010;11(4):413–31.

[17] Ahmad M, Shabbir S, Raza RA, Mazzara M, Distefano S, Khan AM. Hyperspectral image classification: Artifacts of dimension reduction on hybrid CNN. 2021, arXiv preprint arXiv:2101.10532.

[18] Piao J, Chen Y, Shin H. A new deep learning based multi-spectral image fusion method. Entropy 2019;21(6):570.

[19] Sinha A, Shekhawat RS. Olive spot disease detection and classification using analysis of leaf image textures. Procedia Comput Sci 2020;167:2328–36. http://dx.doi.org/10.1016/j.procs.2020.03.285, URL: https://www.sciencedirect.com/science/article/pii/S1877050920307511, International Conference on Computational Intelligence and Data Science.

[20] Kuricheti G, Supriya P. Computer vision based turmeric leaf disease detection and classification: a step to smart agriculture. In: 2019 3rd International conference on trends in electronics and informatics (ICOEI). IEEE; 2019, p. 545–9.

[21] Al Haque AF, Hafiz R, Hakim MA, Islam GR. A computer vision system for guava disease detection and recommend curative solution using deep learning approach. In: 2019 22nd International conference on computer and information technology (ICCIT). IEEE; 2019, p. 1–6.

[22] Rajesh B, Vardhan MVS, Sujihelen L. Leaf disease detection and classification by decision tree. In: 2020 4th International conference on trends in electronics and informatics (ICOEI)(48184). IEEE; 2020, p. 705–8.

[23] Alruwaili M, Alanazi S, Abd El-Ghany S, Shehab A. An efficient deep learning model for olive diseases detection. Int J Adv Comput Sci Appl 2019;10(8).

[24] Alshammari H, Gasmi K, Ben Ltaifa I, Krichen M, Ben Ammar L, Mahmood MA. Olive disease classification based on vision transformer and CNN models. Comput Intell Neurosci 2022;2022.

[25] Fazari A, Pellicer-Valero OJ, Gómez-Sanchıs J, Bernardi B, Cubero S, Benalia S, Zimbalatti G, Blasco J. Application of deep convolutional neural networks for the detection of anthracnose in olives using VIS/NIR hyperspectral images. Comput Electron Agric 2021;187:106252.

[26] Lachgar M, Hrimech H, Kartit A, et al. Optimization techniques in deep convolutional neuronal networks applied to olive diseases classification. Artif Intell Agric 2022;6:77–89.

[27] Alamsyah D, Fachrurrozi M. Faster R-CNN with inception V2 for fingertip detection in homogenous background image. J Phys Conf Ser 2019;1196:012017. http://dx.doi.org/10.1088/1742-6596/1196/1/012017.

[28] Wu Y, Kirillov A, Massa F, Lo W-Y, Girshick R. Detectron2. 2019, https://github.com/facebookresearch/detectron2.

[29] Li K, Lin J, Liu J, Zhao Y. Using deep learning for Image-Based different degrees of ginkgo leaf disease classification. Information 2020;11(2):95.

[30] Ahmad I, Hamid M, Yousaf S, Shah ST, Ahmad MO. Optimizing pretrained convolutional neural networks for tomato leaf disease detection. Complexity 2020;2020:1–6.

[31] Wu M, Yue H, Wang J, Huang Y, Liu M, Jiang Y, Ke C, Zeng C. Object detection based on RGC mask R-CNN. IET Image Process 2020;14(8):1502–8.

[32] Soria C. MM, R. C, J. S. Control of an autonomous all terrain vehicle. In: 8° workshop robot de exteriores ROBOCITY2030. Centro de Automática y Robótica CSIC-UPM; 2010, p. 115–26.

[33] Pedro D. BR, Carlos M. S, Ricardo O. CA. Adaptive configuration arm for the application of foliar actuation liquid treatments. In: 2018 Argentine conference on automatic control (AADECA). 2018, p. 1–6. http://dx.doi.org/10.23919/AADECA.2018.8577372.