Brazilian Journal of Experimental Design, Data Analysis and Inferential Statistics

Federal University of Rio de Janeiro

BJEDIS, Rio de Janeiro, Special Edition, v. 01 (2), 2021

DOI: https://doi.org/10.55747/bjedis.v1i2.48365

ISSN: 2763-6925

Special Edition **Received 03/12/2021. Revised 03/30/2021. Accepted 04/06/2021. Published 07/01/2021.**

Pre-Harvest Fruit Image Processing: A Brief Review

Matheus Felipe Gremes*[1], Rafael Krummenauer[1], Cid Marcos Gonçalves Andrade[1] and Oswaldo Curty da Motta Lima[1]

[1] *Department of Chemical Engineering, State University of Maringá – UEM, Maringá, Brazil.*

Abstract: Agriculture is essential for the development of human civilization. Methods that can precisely estimate the yield of a crop or to perform the harvest automatically using robots can decrease the costs involved and increase production efficiency. With the advancement of agriculture 4.0, current technologies like the internet of things, big data, and artificial intelligence have become more and more common. Systems that use image processing with Deep Learning methods are becoming viable in solving agricultural problems. Deep Learning is part of a large family of methods based on artificial neural networks that can mimic the human brain's work in data processing and pattern recognition for decision-making. Indeed, applications of Deep Learning techniques in agriculture are relatively recent. However, with the rapid advance in Deep Learning and its successful application in agriculture, many articles have been published in recent years. Thus, the main objective of this work was to carry out a brief bibliographic review of pre-harvest fruit image processing techniques, emphasizing the most recent applications using Deep Learning. As seen in the literature, Deep Learning is a promising tool for various agricultural activities, including fruit counting and automatic fruit harvesting using robots.

Keywords: Agriculture, Deep Learning, Image Processing, Fruits.

Adherence to the BJEDIS' scope: This work is closely related to the scope of BJEDIS as it presents a brief review of pre-harvest fruit image processing utilizing Deep Learning methods. Deep learning methods allow the analysis of extremely non-linear relationships in large databases with no known distribution.

* Correspondence address for this author ot the Department of Chemical Engineering, State University of Maringá, City: Maringá, Country: Brazil; Tel: +55-44-99997-8543; E-mails: pg402992@uem.br

1. INTRODUCTION

Agriculture has become increasingly crucial for humanity. Fruits, rich in nutrients, are usually present in the diet of most people. Thus, continuous production is necessary to meet global demand [1]. Producers have always sought ways to increase production quality and decrease costs. One way to achieve this goal is through technological innovations such as computer vision that has shown significant advances in recent years. Computer vision is the science and technology that extracts meaning from images or data and, together with other information processing methods, produces data and insights that help in decision-making [2]. Neural networks with multiple layers, also called Deep Learning, when applied to computer vision, is one of the technological tools that have been increasingly used in the agro-industry, both in automatic fruit harvesting and in fruit selection machines [3].
One of the main objectives of deep learning is to solve tasks that people intuitively solve, but that is not easy to implement computationally [4]. Thus, systems using Deep Learning must have the ability to acquire their knowledge, extracting patterns from raw data, which is known as machine learning [5].
An essential feature of Deep Learning is that this approach achieves high levels of abstraction and pattern recognition present in images. The Convolutional Neural Network (CNN) is the prominent architecture of Deep Learning used in image processing [4].
One of the main characteristics of Deep Learning methods is that they allow the analysis of extremely non-linear relationships in large databases with no known distribution. It is possible to notice a similarity between inferential statistics and Deep Learning; both learn from data representing a part of the whole population and make predictions about the rest of it [6]. However, inferential statistics make an inference about the population based on a sample. In contrast, Deep Learning methods are used to make repeated predictions when finding patterns in the database.
However, Deep Learning techniques applied to agriculture are very recent, as shown in the 2018 review article [7], which reported 40 studies that used Deep Learning somehow in their research. Among these, only 4 used Deep Learning techniques to detect and count fruit in the field.
Another vital article to cite is the 2019 review article [8], which presented a review of the Deep Learning field's rapid development in agriculture - emphasizing the practical aspect of applying models that use Deep Learning in the task of detecting and locating fruits.
Although both of these review articles are recent, the application of Deep Learning in fruit image processing has become so popular that many studies have been published in the past two years. Thus, the main objective of this work was to conduct a brief bibliographic review of the main articles published in 2019 and 2020 that used Deep Learning techniques in the processing of pre-harvest fruit images.
The main characteristics that were considered for the selection of the articles that integrate this bibliographic review were: quality of the obtained results, if the articles obtained relevant results when compared with other works; whether the articles developed a database that could be useful for new works; whether the articles used any innovative approach when compared to other works; and whether the articles used any new deep learning architecture when compared to other works.

1.1. Convolutional neural network

A convolutional neural network is a neural network that has at least one convolution layer instead of fully connected layers [4]. A convolutional neural network tends to demand a minimum level of pre-processing when compared to other classification algorithms. In general, a convolutional neural network adapts a set of filters during the training process, which in other traditional classification algorithms would have to be implemented manually [9].
Each layer of a CNN applies a different set of filters, typically hundreds or thousands of filters, and combines the result by applying them to the next layer's input [9].
In the context of image classification, a CNN learns to detect edges and vertices in the first layers; use these edges and vertices to detect simple shapes in the subsequent layers; use these shapes to detect complex structures like a person's face or parts of a car in the last layers. In the last layer of a CNN, these complex structures are used to make predictions about the image's content [9].

As shown in Figure 1, convolutional neural networks have three different layers: convolutional layers, pooling layers and fully connected layers.

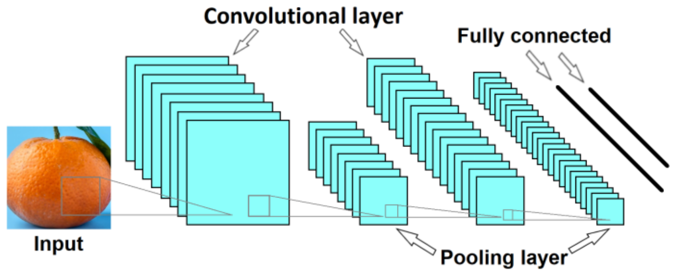

Figure 1: CNN architecture
Source: [10] modified by the author

Unlike a standard neural network, CNN has its layers arranged in volumes with three dimensions: width, length, and depth, where depth refers to the third dimension of the volume, such as the number of channels in an image or the number of filters in a layer [11].

1.2. Convolutional layer

Kernel, or filter, is an array of weights. The kernel traverses the input matrix through a process that implements a sum of the Hadamard product (element-wise product) between the array of weights and the input matrix's values; this operation is called convolution. The kernel always has the same depth dimension as the input matrix, and the size of its spatial dimensions can be controlled by filling in zeros (zero-padding) and the step parameter (strides) [4]. In Figure 2, we illustrate the convolution of a kernel with an input matrix of a dimension of depth one.

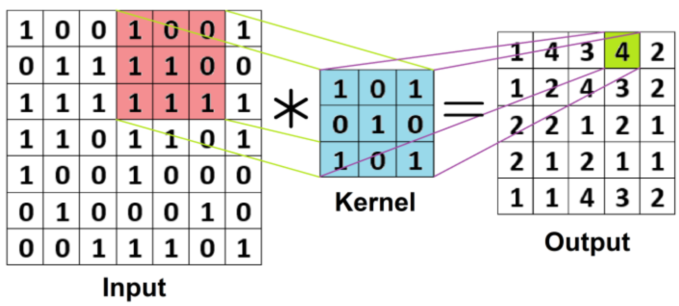

Figure 2: Illustration of the convolution between a 3x3 filter with an input image of depth 1

The convolutional layer consists of filters, or kernels, that are applied along the spatial dimensions and added together over the input volume's depth dimension [11]. The matrix resulting from a convolution operation between a kernel and an input matrix is called an activation map. In Figure 3, it is possible to observe the operation of convolution of an input volume of depth 3 with a filter of spatial dimension 3x3.

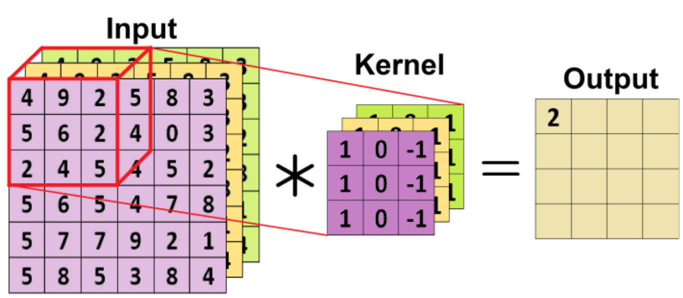

Figure 3: Convolution between a 3x3 filter with a three-channel RGB image

The output matrix that is obtained from this operation is then passed through an activation function. The function commonly used is ReLu and can be calculated by.

$$f(x) = \max(0, x) \tag{1}$$

The ReLu function makes the learning convergence of CNN networks faster and at a lower computational cost when compared to functions used initially, such as sigmoids [12].

1.3. Pooling layer

A pooling layer is usually added after the convolutional layer. The pooling layer is responsible for reducing the input volume's spatial dimension, thereby reducing the number of parameters to be trained [11].
The most commonly applied pooling technique is to replace a region's values with the maximum value contained in that region, as shown in Figure 4.

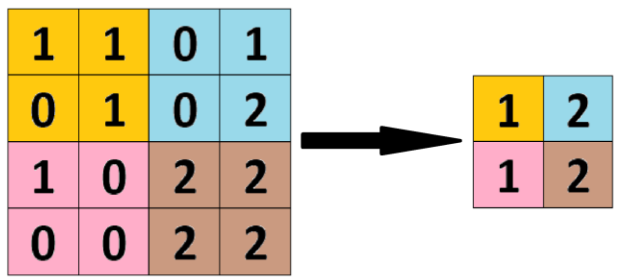

Figure 4: Application of max-pooling in a 4x4 image using a 2x2 filter

1.4. Fully connected layer

After the convolutional layer and the pooling layer, the last activation map used for the classification task is processed using the fully connected layer. The fully connected layer can receive only one-dimensional data. In this way, the data organized in a volume with three dimensions of the last layer needs to be converted to unitary dimension [11].
All neurons in the fully connected layers are connected with all neurons in the previous layer, as shown in Figure 5. Furthermore, they work the same way as in a conventional neural network.

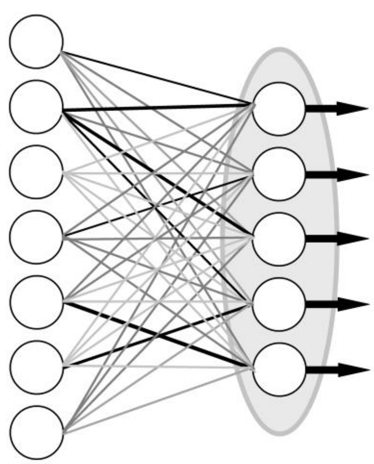

Figure 5: Fully connected layer

1.5. Classification, Detection, and Segmentation of images

There are three most popular applications from the perspective of computer vision involving convolutional neural networks: classification, detection, and segmentation of images.
Image classification is the task of labeling an entire image as an object or concept [12]. Figure 6 shows an example.

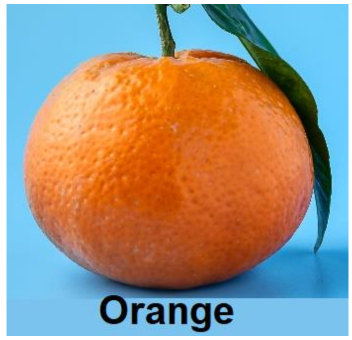

Figure 6: Image classified as Orange
Source: [10] modified by the author

Image detection is the task of detecting and demarcating bounding boxes on objects in an image [11]. Figure 7 shows the detection of three oranges.

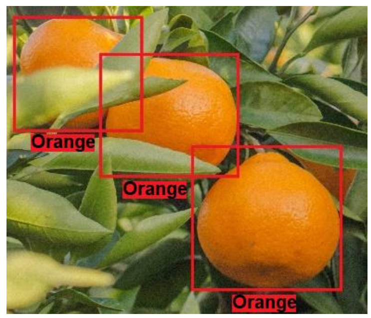

Figure 7: Image detection in a photo with three oranges
Source: [13] modified by the author

Image segmentation is the pixel-based sorting task. There are two primary image segmentation forms: semantic segmentation [14] and instance segmentation [15]. In semantic segmentation, each pixel of the image will be classified as belonging to a class; regardless of whether there is more than one object with the same class, the semantic segmentation will not distinguish between the objects as distinct entities. In instance segmentation, additionally to the classification of each pixel as belonging to a class, an instance will also be assigned to each object; thus, the instance segmentation separates the objects as distinct entities, even if they have the same class. Figure 8 shows an example of semantic segmentation and instance segmentation.

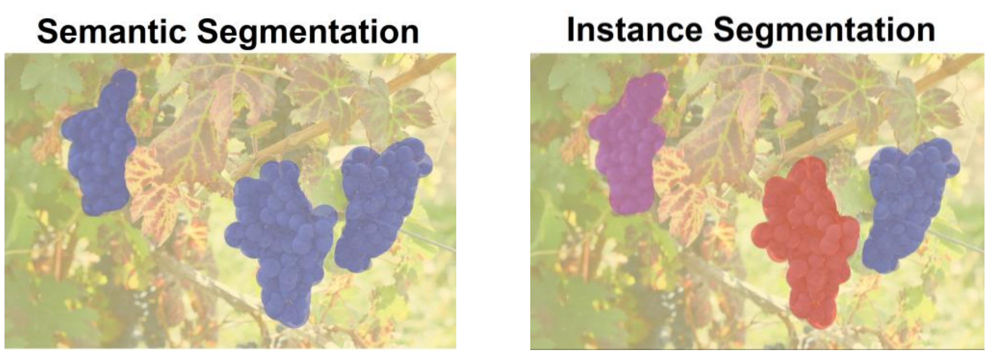

Figure 8: Semantic segmentation and instance segmentation in grape clusters
Source: [16] modified by the author

2. PRE-HARVEST FRUIT IMAGE PROCESSING

This section presents two tables containing the most recent and relevant articles that used Deep Learning techniques for image processing of pre-harvest fruits. Table 1 shows the studies that used techniques to perform fruit segmentation; Table 2 shows the studies that used techniques to detect and demarcate the fruits with bounding boxes. A brief explanation of the objectives, methods, and results obtained are given for each article presented in the tables.

Table 1. Works that used segmentation techniques in the processing of pre-harvest fruit images.

Fruit	Main method	Data type	Main objectives	Performance Results	Speed / s	Ref
Mangos	VGG nets modified	RGB Images 200 x 200	Detection and counting of mangos	Accuracy 73.6% F1-score 84.4%	-	[17]
Apples	U-net modified	RGB Images 224 x 224	Detection and counting of apples	Counting accuracy 95.56% - 97.83%	-	[18]
Strawberries	Mask-RCNN	RGB Images 640 x 480	Detection for use in harvesting robots	Accuracy 95.78%	-	[19]
Cucumbers	Mask-RCNN modified	RGB Images 600 x 400	Detection of cucumbers	Accuracy 90.68% F1-score 89.47%	0.3461	[20]
Strawberries	Mask-RCNN	RGB-D Images 640 x 480	Detection and segmentation of strawberries for harvest using a robot	Yield accuracy of the crop 74.1%	0.62	[21]
Strawberries and twigs	DaSNet modified	RGB-D Images 320 x 320	Detection and segmentation of apples and branches	F1-score 83.2%	0.032	[22]
Lychees and twigs	YOLOv3 and U-net	RGB Images 150 x 150	Detection and segmentation of lychees and branches in night environments	Detection accuracy of Lychees 96.43% Segmentation accuracy of Branch 95.54%	0.097	[23]
Apples	FCF neural network based on ResNet-50	RGB Images 404 x 303	Apple growth monitoring system	-	-	[24]
Apples	Mask R-CNN	RGB Images 1024 x 1024	Fruit detection and 3D location using SfM	F1-score 88.1%	-	[25]
Apples	DaSNet	RGB-D Images	Detecting fruits and determining the best position for the robotic arm to harvest	F1-score 87.3% Harvest rate 84.7%	-	[26]
Bunches of grapes	Mask R-CNN	RGB Images 1024 x 1024	Detect, segment and track bunches of grapes	F1-score 91.0%	-	[27]
Apples and twigs	DaSNet modified. DaSNet-v2	Images RGB-D 640 x 480	Detect and segment apples and branches	Segmentation accuracy of fruits 86.6% Segmentation accuracy of Branch 75.7%	0.055	[28]
Green Apples	Mask-RCNN modified	RGB Images 512 x 512	Detect and segment overlapping fruits	Accuracy 97.31% Recall Rate 95.70%	-	[29]
Lychee branches	DeepLabV3+ model	RGB Images	Segmentation of lychee branches	MIoU 0.765	-	[30]

Let the analyses of the selected contributions in the related literature be presented in the same order as organized in Table 1.

The work conducted by Kester and Meduri [17] proposed a detector and mango counter for operating in open fields, using a deep network developed for this project, called MangoNet, that uses semantic segmentation. The architecture of the MangoNet network was inspired by the VGG network [31]. The final layer of the MangoNet network produces a segmented image in pixels that are regions expected to belong to the mango class; these regions are used for the detection and counting of fruits. The proposed detection method proved very robust to variations in scale, lighting, and occlusion compared to similar architectures.

Using deep learning methods combined with the semi-supervised method based on GMM (Gaussian Mixture Models) [32], Nicolai and Pravakar [18] achieved an estimate of fruit production with precision ranging from 95.56%

to 97.83%. The authors presented a semantic segmentation network based on the U-Net architecture [33]. The neural network presented a better performance in all the fruit counting tests performed when compared to the model based on GMM. Interestingly, when both methods were combined, the obtained precision was superior to each one of the isolated methods.

In order to improve the performance of computer vision machines in detecting strawberries for harvesting robots, Yu and Zhang [19] developed a network architecture capable of detecting ripe and unripe strawberries, based on the Mask Region Neural Network (Mask R-CNN) [34] using the convolutional neural network Resnet-50 as a backbone [9], combined with the Feature Pyramid Network (FPN) architecture [35] to extract features. The proposed method was compared with four other traditional methods and showed better results, being effective in situations of overlap, occlusion, and variation of light. The method obtained an average detection accuracy of 95.78%, but with a slow execution speed for real-time applications caused by the large number of calculations performed in the deep neural network.

The work conducted by Liu and Zhao [20] was also based on the Mask R-CNN architecture [34], but the authors have chosen to use Resnet-101 [9] as a backbone and Feature Pyramid Network (FPN) [35] for extracting features. The work's objective was to detect cucumbers in the growing environment, a task considered difficult due to the similarity of fruit's color with the foliage. The proposed model obtained better performance in the F1-score metric 89.47% compared with the other four detection methods based on deep learning. However, the model obtained an average execution time of 0.3461s, which may be not enough for real-time applications depending on the employed hardware. The high time is due to the Mask R-CNN structure [34] that is composed of two stages.

A computer vision system for detecting and locating strawberries was presented by Ge and Xiong [21] for use in a harvesting robot to collect table strawberries. The Mask R-CNN [34] architecture was used to detect strawberries through images from an RGB-D camera, a camera capable of providing information about the depth of objects in the image. The 3D location of the objects is done through an algorithm that performs a transformation of coordinates from 2D images. The tests showed a harvest accuracy of 74.1% in the proposed situations. The work investigated the challenges of the fruits' location based on deep neural networks of segmentation. It raised some problems of the perception of the harvest environment, presenting methods for detecting the objects for better decision-making in the safe manipulation of the harvest robot.

Kang and Chen [22] proposed a multifunctional neural network to perform the real-time semantic detection and segmentation of strawberries and twigs in natural environments using a Kinect-V2 camera, a RGB-D visual sensor. The developed multifunctional neural network was named Detection and Segmentation Network (DaSNet), which was devised to exploit as feature extractors the Gated Feature Pyramid Network (GFPN) and the Atrous Spatial Pyramid Pooling (ASPP). The DasNet network with ReSNet-101 [9] as a backbone obtained the best performance compared to other object detectors with an F1-score of 83.2% in detecting apples and 87.6% and 77.2% in the accuracy of segmentation of apples and branches, respectively. The system performed detection and segmentation in real-time when a lighter neural network was used as a backbone, reaching an execution time of 0.032s. However, the F1-score obtained was 82.1% in the detection of apples and 86.8% and 75.7% in the segmentation precision of apples and branches, respectively.

In another work [23] a method was proposed to detect lychees and their branches in night environments. The object detector was based on YOLOv3 [36]. The regions of interest for the lychee branches were determined based on bounding boxes of fruits provided by YOLOv3. Finally, the lychee branches were segmented using an architecture based on the U-Net neural network [33]. The experiments carried out in the work showed that the average accuracy in the detection was 96.78%, 99.57%, and 89.30% under conditions of high illumination, normal illumination, and low illumination, respectively. In comparison, the Mean Intersection over Union (MIoU) of the lychee branches' segmentation was 79.00%, 84.33%, and 78.60% under the same conditions, respectively. The system execution time to perform object detection and segmentation was 0.097s.

With the general objective of developing an apple growth monitoring system in an orchard based on a deep neural network of edge detection for remote estimation of apple size during the entire growing period, Wang and Li [24] built the system using a spherical camera and two personal computers. The edges detection neural network developed for fruit segmentation was the Fused Convolutional Features (FCF) based on the ResNet-50 neural network [9]. The system was able to monitor the growth of 21 apples in 3 different trees.

Another work that performed detection and localization of fruits in 3D environments was proposed by Gené-Mola and Sanz-Cortiella [25]. The neural network used to segment the apples in the orchards was the Mask R-CNN [34], and the technique used to project the objects in a 3D space was the structure-from-motion (SfM) [37]. The results showed that the combination of segmentation with SfM increased the performance of F1-score fruit detection from 81.6% to 88.1% concerning the total number of detected fruits. The main advantage of the method was the reduction in the number of false positives. However, the most significant disadvantage was the processing time required by the SfM, which makes it impossible to execute the system in real-time.

Kang and Chen [26] proposed a system that includes a multifunctional neural network to detect apples and a Pointnet neural network [38] to determine the best position for the robotic arm to harvest the fruit. The detection and segmentation of the fruits are performed through images acquired by an RGB-D camera. The chosen architecture was the neural network of a DaSNet stage with the neural network resnet-50 [9] as a backbone. The whole system was tested together with a robotic arm in a controlled environment and obtained a harvest success rate of 84.7% and an F1-score of 87.3% in detecting apples.

The bunches of grapes vary widely in shape, color, and size, making detecting them a difficult task. However, Santos and Souza [27] demonstrated that bunches of grapes could be successfully detected, segmented, and tracked using the latest generation convolutional neural networks. Three neural networks were trained and evaluated in this work: Mask R-CNN [34], YOLOv2 [39], and YOLOv3 [36]. The authors demonstrated that the bunches of grapes could be identified and tracked over a sequence of video frames recorded online in the vineyard. It has also been shown that 3D models produced by structure-from-motion (SfM) [37] can be used to avoid double curl counting, thus reducing detection errors. F1-scores greater than 90% were obtained in the tests performed.

Kang and Chen [28] presented a deep neural network, which can perform detection and segmentation of apples, and semantic segmentation of branches, showing promising results in detecting overlapping fruits. For DaSNeT-v2 using Resnet-101 [9] as a backbone, an average accuracy of 88% in fruit detection and a segmentation precision of 87.3% were obtained with an execution time of 0.070s. In comparison, DaSNeT- v2 using Resnet-18 [9] as a backbone obtained an average accuracy of 87% in detecting fruits and a segmentation precision of 86.6% with an execution time per image of 0.054s.

The detection of greenish-colored fruits is generally more complex than that of other fruits because their colors are similar to those of the foliage. Jia and Tian [29] presented a detection model based on the Mask Region Convolutional Neural network (Mask R-CNN) architecture [34]. The model was developed to be effective in the detection and segmentation of overlapping fruits. The neural networks Residual Network (ResNet) [9] and Densely Connected Convolutional Networks (Dense Net) [40] were combined and used as a backbone for the extraction of characteristics. The method was tested on a set of 120 images, obtaining an average detection accuracy rate of 97.31% and a Recall Rate of 95.70%. It also showed promising results in detecting fruits partially hidden by the foliage.

In order to build a convolutional neural network capable of semantic segmenting lychee branches, Peng and Xue [30] used the DeepLabV3 + [41] model combined with the Xception neural network [42] as a feature extractor. A coding and decoding structure was used to reduce parameters in the neural network. The coding framework removes the pooling layers from the feature extraction network to keep the high-level abstract information large enough to facilitate pixel location prediction. Meanwhile, the decoding framework uses 1x1 dimension filters to reduce the number of layers in the activation map. The model, which locates branches through semantic segmentation, obtained a value of 0.765 in the Mean intersection over union (MIoU) parameter.

Table 2: Works that used object detection techniques in the processing of pre-harvest fruit images.

Fruit	Main method	Data type	Main objectives	Performance Results	Speed /s	Ref
Mangos	Mango YOLO based on YOLOv3	RGB Images 512 x 512	Detection and counting of mangos	F1-score 97.0%	0.015	[43]
Apples	YOLO modified	RGB Images 1216 x 1216	Detection and counting of apples	F1-score 79.0%	0.050	[44]
Kiwis	Faster R-CNN	RGB-D and NIR Images	Kiwi Detection	Accuracy 90.7%	0.135	[45]
Oranges	Faster R-CNN modified	RGB Images 480 x 270	Small fruits detection	F1-score 86.78%	0.085	[46]
Apples	LedNet	RGB-D Images 640 x 480	Develop a structure that includes a label generation module and a real-time one-stage detector	F1-score 84.9% Accuracy 86.4%	0.046	[47]
Apples Mangos Oranges	Faster R-CNN modified	RGB Images 100 x 100	Develop a multiclass fruit detector and create a database with images of the fruits	Accuracy 90.72%	0.058	[48]
Oranges	Faster R-CNN	RGB Images 600 x 600	Develop a methodology to detect, count and estimate the size of oranges using an unmanned aerial vehicle	Crop production estimate error 7.22%	-	[49]

Tomates	Faster R-CNN	RGB Images 1296 x 864	Detect green, intact tomatoes regardless of occlusions or of the fruit growth stage	Accuracy 87.83%	0.37	[50]
Green mangoes	YOLOv2	RGB Images 544 x 544	Develop a method for detecting and counting green mangos using an unmanned aerial vehicle	Accuracy 96.1%	0.08	[51]
Avocados Apples Lemons	Faster R-CNN and Single Shot Multibox Detector (SSD)	RGB Images 360 x 640	Test two common Faster R-CNN and SSD architectures	Counting accuracy 7%, 13% and 20% with Faster R-CNN Counting accuracy 18%, 10% e 25% With SSD	Faster R-CNN 0.220 SSD 0.060	[52]
Apples	Faster R-CNN	RGB Images 416 x 416	Detect and estimate the number of fruit crop production using an unmanned aerial vehicle	Counting accuracy 88.96%	-	[53]
Apples	Faster R-CNN	RGB-D Images 224 x 224	Develop an apple detection system using a Kinect V2 sensor to improve detection accuracy	Accuracy 89.3%	0.181	[54]
Apple Blossoms	YOLOv4 modified	RGB Images 416 x 416	Detect apple blossoms in real-time	Accuracy 97.31%	0.014	[55]

Koirala and Walsh [43] compared the performance of six existing deep learning architectures in the task of detecting mangoes directly from trees. The trees' images were obtained from five mango plantations using a 5 Megapixel RGB digital camera in a vehicle moving at 6 km/h. The used architectures were the two-stage detectors Faster R-CNN (VGG) and Faster R-CNN (ZF) [56], and the one-stage detectors YOLOv3 [36], YOLOv2, YOLOv2 (tiny) [39], and SDD. A new architecture was also developed, called MangoYOLO, based on YOLOv3 and YOLOv2 (tiny) characteristics. MangoYOLO obtained the best performance compared to the other methods obtaining an F1-score of 96.7% and an average detection precision of 98.6%. The model's execution time was 0.015s, second only to YOLOv2 (tiny). MangoYOLO obtained a production estimate error between 4.6% and 15.2% in the plantations in which it was applied.

Bresilla and Perulli [44] dealt with convolutional neural network architectures based on one-stage detectors of apples directly from orchards. Inspired by the YOLO architecture [57], the authors' model demonstrated an accuracy of detection of more than 90%. Based on the correlation between the actual number of fruits in a tree, the number of fruits detected in a frame, and the number of visible fruits, a model was created to accommodate this error rate. The processing speed obtained was just over 20 FPS. The changes made to the YOLO architecture turned it more accurate in detecting objects of the same class when they are very close to each other.

The work conducted by Liu and Wu [45] presented a method for detecting kiwi fruits by applying RGB-D sensors and using RGB images and Near-Infrared (NIR) images as input to a modified layer of the Faster R-CNN architecture [56] with the neural network VGG16 [31] as backbone to receive an image with six channels. Two different methods of extracting features were used: Image-Fusion and Feature-Fusion. The tests demonstrated that the average precision and detection time by image using the original VGG16 with only the RGB and NIR images as input were only 88.4% and 0.134s; 89.2% and 0.134s, respectively. Using VGG16 with the Feature-Fusion method, an average accuracy of 90.5% was obtained and an image detection time of 0.188s, while using the Image-Fusion method, an average accuracy of 90.7% was achieved with a time of 0.135s image detection.

Small objects and fruits are generally more difficult to detect in an image. To make the detection of small fruits more effective, Mai and Zhang [46] proposed the Faster R-CNN [56] architecture with a fusion classifier. The system was evaluated into two sets of small fruit images and obtained the best F1-score, 86.78%, compared to other object detectors. The image processing time was 0.085s. The work's main contribution was to present strategies on how to increase the performance of neural networks in the detection of small fruits.

Methods based on deep learning generally require training data appropriately labeled. Labeling those data can be a time-consuming task. Kang and Chen [47] proposed a system for detecting apples in orchards that comprises an automatic labeling module and a LedNet stage detector. The automatic labeling module uses a multi-scale pyramid and a clustering-RCNN (C-RCNN) to assist in the rapid labeling of training data, while LedNet uses the Feature Pyramid Network (FPN) [35] and the Atrous Spatial Pyramid Pooling (ASPP) to improve detection performance. LedNet using resnet-101 [9] as a backbone reached an F1-score of 84.9% with an image processing time of 0.046s.

Wan and Goudos [48] proposed a multi-class fruit detector based on the Faster R-CNN architecture [56] for detecting apples, mangoes, and oranges. They were also responsible for creating a dataset containing labeled images of the mentioned fruits. The neural network was tested and compared with other models and obtained a performance of 90.72% in the average detection precision and an image processing time of 0.058s. The most significant contribution of this work was creating a fruit image dataset and optimizing the structure of the Faster R-CNN.

The main objective of the work conducted by Apolo-Apoloa and Martínez-Guanter [49] was to develop a methodology to detect, count, and estimate the size of the oranges on trees using deep learning techniques. A Faster R-CNN neural network [56] was trained using images of the oranges photographed by an unmanned aerial vehicle. An average error between the visual count and the count performed by the model of 6.59% was obtained. However, the most important result is the production estimate for the year: the average error found for this case using the proposed model was 7.22%, while the error found by technical experts was 13.74%.

Mu and Chen [50] developed a model to detect green tomatoes regardless of the level of occlusion or the stage of growth. The three models developed for this work used the Faster R-CNN [56] architecture using as backbone Resnet-50, Resnet-101, and Inception-Resnet-v2 [9]. The model that obtained the highest performance was the one that used the Resnet-101 neural network, with an average detection accuracy of 87.83%. In counting tomatoes, the model received a high coefficient of determination (R2 = 0.87), with more than 10% of the tomatoes in the images considered small (with size less than 50 pixels).

The YOLOv2 architecture [39] was used to detect and count mangos in images photographed by an unmanned aerial vehicle in the work conducted by Xiong and Liu [51]. The images were manually labeled. Good results were obtained in images containing different numbers of fruits and under different lighting conditions, reaching an average detection accuracy of 96.1% with an image processing time of 0.08s. A model for estimating the number of mangos was obtained by linear adjustment between the actual number and the number of detected mangos. The model reached an estimate error of 1.1% in the number of mangos of 10 trees. The validation of the error was determined by manual counting.

In another work [52], a comparison is made in the task of detecting and counting avocados, apples, and lemons between two popular architectures: Faster R-CNN [56] with Inception V2, and Single Shot Multibox Detector (SSD) [58] with MobileNet. The methodology for counting fruit consisted of recording video images of the orchards and later processing them for counting the fruits. A counting error of 7%, 13%, and 20% was obtained for avocados, apples, and lemons, respectively, using the Faster R-CNN architecture with an image processing time of 0.220s. In contrast, for the SSD architecture, the counting error obtained was 18%, 10%, and 25% for avocados, apples, and lemons, respectively, with an image processing time of 0.060s.

The study developed by Apolo-Apolo and Pérez-Ruiz [53] presents a method for detecting and counting apples in orchards to estimate production using images obtained by an unmanned aerial vehicle. The Faster R-CNN architecture [56] with the Inception Resnet V2 Atrous Coco neural network was chosen to detect fruits. The Faster R-CNN neural network was trained using the Google Collaboratory platform, a cloud service based on Jupyter Notebooks, which allows integrating deep learning models in a simple Python script. The results obtained by the model were compared with the count performed by specialist technicians. As fruits were partially visible in the images obtained by the unmanned aerial vehicle, a linear regression was used to estimate the total number of apples in each tree, obtaining an R2 equal to 0.80.

Fu and Majeed [54] developed a computer vision system using a low-cost Kinect V2 sensor for apple detection. Two architectures were developed using Faster R-CNN with ZFNet and Faster R-CNN with VGG16 [56], [31]. Both are used in the original RGB images and the RGB images with suppressed background. The authors' tests demonstrated that Faster R-CNN obtained the highest average precision of 89.3% with VGG16 with an image processing time of 0.181s. The results showed that using a depth filter to suppress the background of the original images improved the detection accuracy by 2.5%.

Wu and Shuaichao [55] developed a method for real-time detection of apple blossoms using YOLO v4 [59] pruned by a channel pruning algorithm. After pruning, the number of parameters was reduced by 96.74%, causing a reduction of the model in 231.51MB. The processing time per image was reduced by 39.47%, remaining at 0.014s, while the average detection precision remained at 97.31%, which is only 0.24% less than in the model before pruning. The tests also showed that the different species of orchards and the difference in lighting did not significantly impact the detection of apple blossoms.

It is possible to notice that the obtained results in the related literature are improving in comparison with older articles. It is probably due to both new and better network architectures and also more significant amount of data available for training those networks. It is a growing tendency for the authors to make available the dataset created and used for training their solutions, so it is reasonable that in the perspective of pre-harvest fruit image processing, the quantity and quality of properly labelled images available online shall increase over time. Thus, we believe that this movement around the datasets tends to make the comparison of the performance of different network architectures to be tight and easier.

Some similarities are also observed in the strategies of some authors. When the objective is to carry out fruit counting, the authors tend to opt for detection using bounding boxes. However, when the objective involves knowing the fruit's position and shape with greater precision, the authors opt for segmentation techniques.

3. CONCLUSION

In this review work, proposals and contributions from recent works published in the last two years representing state of the art in Deep Learning techniques applied in image processing of pre-harvest fruits were analyzed and compared. As seen in previous reviews, applications of Deep Learning in image processing in agriculture are very recent. Besides, it should be noted that most of these studies collect information before 2019. We note that the architectures of approaches based on Deep Learning vary according to different applications, works, and authors. Thus, we cannot establish one architecture as being superior to the others. The results showed that approaches based on Deep Learning achieved excellent results. We can also observe that Deep Learning applications' most significant growths are in the robotic harvesting sector and the production estimation sector.

ACKNOWLEDGMENT

The authors would like to thank the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) for the financial support.

CONFLICT OF INTEREST

There is no conflict of interests.

Sample CRediT author statement

Matheus Felipe Gremes: Conceptualization, preliminary studies, literature search, review, writing - preparing the original draft, writing - proofreading and editing. **Rafael Krummenauer:** Preliminary studies, writing - proofreading and editing. **Cid Marcos Gonçalves Andrade:** Conceptualization, supervision, writing - proofreading and editing. **Oswaldo Curty da Motta Lima:** supervision, writing - proofreading and editing.

REFERENCES

1. ABDULLAHI, Halimatu Sadiyah, SHERIFF, Ray E and MAHIEDDINE, Fatima. Convolution neural network in precision agriculture for plant image recognition and classification 5G Mobile Wireless Network and Technology View project Automation on farm field operations for yield optimization View project Convolution Neural Network in Precision Agriculture for Plant Image Recognition and Classification**. researchgate.net [online]**. DOI 10.1109/INTECH.2017.8102436. Available from: https://www.researchgate.net/publication/320971643. Accessed 22 March 2021.
2. ZHAI, Zhaoyu, MARTÍNEZ, José Fernán, BELTRAN, Victoria and MARTÍNEZ, Néstor Lucas. Decision support systems for agriculture 4.0: Survey and challenges**. Computers and Electronics in Agriculture [online]**. v. 170. 2020. DOI 10.1016/j.compag.2020.105256. Available from: https://www.sciencedirect.com/science/article/pii/S0168169919316497. Accessed 22 March 2021.
3. TAO, Y, AGRICULTURE, J Zhou - Computers and electronics in and 2017, undefined. Automatic apple recognition based on the fusion of color and 3D feature for robotic fruit picking**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S0168169917302764. Accessed 22 March 2021.
4. GOODFELLOW, I, BENGIO, Y, COURVILLE, A and BENGIO, Y. Deep learning**. [online]**. 2016. DOI 10.4258/hir.2016.22.4.351. Available from: https://doi.org/10.4258/hir.2016.22.4.351. Accessed 22 March 2021.
5. AND, Jones and PUBLISHERS, Bartlett. *Artificial Intelligence Illuminated Ben Coppin* [online]. 2004. Available from: www.jbpub.com. Accessed 22 March 2021.
6. GOPAL, M. Applied machine learning**. [online]**. 2019. Available from: https://www.accessengineeringlibrary.com/content/book/9781260456844. Accessed 22 March 2021.
7. KAMILARIS, Andreas and PRENAFETA-BOLDÚ, Francesc X. *Deep Learning in Agriculture: A Survey*

[online]. [no date]. Available from: https://www.sciencedirect.com/science/article/pii/S0168169917308803. Accessed 22 March 2021.

8. KOIRALA, A, WALSH, KB, WANG, Z, IN, C McCarthy - Computers and electronics and 2019, undefined. Deep learning–Method overview and review of use for fruit detection and yield estimation**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S0168169919301164. Accessed 22 March 2021.

9. ADRIAN, Rosebrock. Deep Learning for Computer Vision with Python - Starter Bundle**.** . P. 332. 2017.

10. Orange fruit photo – Free Food Image on Unsplash. [online]. Accessed 22 March 2021. Available from: https://unsplash.com/photos/U1iYwZ8Dx7k.

11. GOLLAPUDI, S. Learn computer vision using OpenCV**. [online]**. 2019. Available from: https://link.springer.com/content/pdf/10.1007/978-1-4842-4261-2.pdf. Accessed 22 March 2021.

12. KRIG, Scott. **Computer Vision Metrics**. Springer International Publishing, 2016.

13. selective focus photography of unpicked orange fruits during daytime photo – Free Image on Unsplash. [online]. Accessed 22 March 2021. Available from: https://unsplash.com/photos/4XoCxdMnWFg.

14. GUO, Yanming, LIU, Yu, GEORGIOU, Theodoros and LEW, Michael S. A review of semantic segmentation using deep neural networks**. International Journal of Multimedia Information Retrieval**. v. 7, n. 2, p. 87–93. 2018. DOI 10.1007/s13735-017-0141-z.

15. HAFIZ, Abdul Mueed and BHAT, Ghulam Mohiuddin. A survey on instance segmentation: state of the art**. International Journal of Multimedia Information Retrieval**. v. 9, n. 3, p. 171–189. 2020. DOI 10.1007/s13735-020-00195-x.

16. Blue berries on brown stem photo – Free Nature Image on Unsplash. [online]. Accessed 22 March 2021. Available from: https://unsplash.com/photos/jzmZeQyEgiU.

17. KESTUR, R, MEDURI, A, ARTIFICIAL, O Narasipura - Engineering Applications of and 2019, undefined. MangoNet: A deep semantic segmentation architecture for a method to detect and count mangoes in an open orchard**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S0952197618301970. Accessed 22 March 2021.

18. NICOLAI, Hni, PRAVAKAR, Roy, HÄNI, Nicolai, ROY, Pravakar and ISLER, Volkan. Author Manuscript A Comparative Study of Fruit Detection and Counting Methods for Yield Mapping in Apple Orchards**. Wiley Online Library [online]**. v. 37, n. 2, p. 263–282. 2020. DOI 10.1002/rob.21902. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21902. Accessed 22 March 2021.

19. YU, Y, ZHANG, K, YANG, L, AGRICULTURE, D Zhang - Computers and Electronics in and 2019, undefined. Fruit detection for strawberry harvesting robot in non-structural environment based on Mask-RCNN**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S0168169919301103. Accessed 22 March 2021.

20. LIU, X, ZHAO, D, JIA, W, JI, W, RUAN, C, ACCESS, Y Sun - IEEE and 2019, undefined. Cucumber fruits detection in greenhouses based on instance segmentation**. ieeexplore.ieee.org [online]**. Available from: https://ieeexplore.ieee.org/abstract/document/8843929/. Accessed 23 March 2021.

21. GE, Y, XIONG, Y, TENORIO, GL, ACCESS, PJ From - IEEE and 2019, undefined. Fruit localization and environment perception for strawberry harvesting robots**. ieeexplore.ieee.org [online]**. Available from: https://ieeexplore.ieee.org/abstract/document/8863343/. Accessed 23 March 2021.

22. KANG, Hanwen and CHEN, Chao. Fruit Detection and Segmentation for Apple Harvesting Using Visual Sensor in Orchards**. mdpi.com [online]**. DOI 10.3390/s19204599. Available from: www.mdpi.com/journal/sensors. Accessed 23 March 2021.

23. LIANG, C, XIONG, J, ZHENG, Z, ZHONG, Z, … Z Li - … and Electronics in and 2020, undefined. A visual detection method for nighttime litchi fruits and fruiting stems**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S0168169919313274. Accessed 23 March 2021.

24. WANG, D, LI, C, SONG, H, XIONG, H, LIU, C, ACCESS, D He - IEEE and 2020, undefined. Deep learning approach for apple edge detection to remotely monitor apple growth in orchards**. ieeexplore.ieee.org [online]**. Available from: https://ieeexplore.ieee.org/abstract/document/8981980/. Accessed 23 March 2021.

25. GENÉ-MOLA, J, SANZ-CORTIELLA, R, … JR Rosell-Polo - … and Electronics in and 2020, undefined. Fruit detection and 3D location using instance segmentation neural networks and structure-from-motion photogrammetry**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S0168169919321507. Accessed 23 March 2021.

26. KANG, Hanwen and CHEN, Chao. *Real-Time Fruit Recognition and Grasping Estimation for Autonomous Apple Harvesting* [online]. [no date]. Accessed 23 March 2021. Available from: https://arxiv.org/abs/2003.13298.

27. SANTOS, Thiago T, DE SOUZA, Leonardo L, DOS SANTOS, Andreza A and AVILA, Sandra. Grape detection, segmentation, and tracking using deep neural networks and three-dimensional association**. Elsevier [online]**. 2020. DOI 10.1016/j.compag.2020.105247. Available from: https://doi.org/10.1016/j.compag.2020.105247. Accessed 23 March 2021.

28. KANG, Hanwen and CHEN, Chao. *Fruit Detection, Segmentation and 3D Visualisation of Environments in Apple Orchards* [online]. [no date]. Accessed 23 March 2021. Available from: https://www.sciencedirect.com/science/article/pii/S0168169919323889.

29. JIA, W, TIAN, Y, LUO, R, ZHANG, Z, … J Lian - Computers and Electronics and 2020, undefined. Detection and segmentation of overlapped fruits based on optimized mask R-CNN application in apple harvesting robot**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S0168169919326274. Accessed 23 March 2021.

30. PENG, H, XUE, C, SHAO, Y, CHEN, K, XIONG, J, … Z Xie - IEEE and 2020, undefined. Semantic Segmentation of Litchi Branches Using DeepLabV3+ Model**. ieeexplore.ieee.org [online]**. Available from: https://ieeexplore.ieee.org/abstract/document/9186684/. Accessed 23 March 2021.

31. SIMONYAN, Karen and ZISSERMAN, Andrew. Very deep convolutional networks for large-scale image recognition. In : **3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings**. International Conference on Learning Representations, ICLR, 2015.

32. MAGAZINE, TK Moon - IEEE Signal processing and 1996, undefined. The expectation-maximization algorithm**. ieeexplore.ieee.org [online]**. Available from: https://ieeexplore.ieee.org/abstract/document/543975/. Accessed 23 March 2021.

33. RONNEBERGER, Olaf, FISCHER, Philipp and BROX, Thomas. U-net: Convolutional networks for biomedical image segmentation. In : **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**. Springer Verlag, 2015. p. 234–241. ISBN 9783319245737.

34. HE, Kaiming, GKIOXARI, Georgia, DOLLÁR, Piotr and GIRSHICK, Ross. *Mask R-CNN* [online]. [no date]. Accessed 23 March 2021. Available from: http://openaccess.thecvf.com/content_iccv_2017/html/He_Mask_R-CNN_ICCV_2017_paper.html.

35. LIN, Tsung-Yi, DOLLÁR, Piotr, GIRSHICK, Ross, HE, Kaiming, HARIHARAN, Bharath and BELONGIE, Serge. *Feature Pyramid Networks for Object Detection* [online]. [no date]. Accessed 23 March 2021. Available from: http://openaccess.thecvf.com/content_cvpr_2017/html/Lin_Feature_Pyramid_Networks_CVPR_2017_paper.html.

36. REDMON, Joseph and FARHADI, Ali. *YOLOv3: An incremental improvement*. 2018. arXiv.

37. SCHÖNBERGER, Johannes L and FRAHM, Jan-Michael. *Structure-from-Motion Revisited* [online]. [no date]. Accessed 23 March 2021. Available from: https://github.com/colmap/colmap.

38. LIANG, Hongzhuo, MA, Xiaojian, LI, Shuang, GÖRNER, Michael, TANG, Song, FANG, Bin, SUN, Fuchun and ZHANG, Jianwei. *PointNetGPD: Detecting Grasp Configurations from Point Sets* [online]. [no date]. Accessed 23 March 2021. Available from: https://lianghongzhuo.github.io/PointNetGPD.

39. REDMON, Joseph and FARHADI, Ali. *YOLO9000: Better, Faster, Stronger* [online]. [no date]. Accessed 23 March 2021. Available from: http://pjreddie.com/yolo9000/.

40. HUANG, Gao, LIU, Zhuang, VAN DER MAATEN, Laurens and WEINBERGER, Kilian Q. *Densely Connected Convolutional Networks* [online]. [no date]. Accessed 23 March 2021. Available from: https://github.com/liuzhuang13/DenseNet.

41. CHEN, Liang-Chieh, ZHU, Yukun, PAPANDREOU, George, SCHROFF, Florian and ADAM, Hartwig. *Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation* [online]. [no date]. Accessed 23 March 2021. Available from: https://github.com/tensorflow/models/tree/master/.

42. CHOLLET, François. *Xception: Deep Learning with Depthwise Separable Convolutions* [online]. [no date]. Accessed 23 March 2021. Available from: http://openaccess.thecvf.com/content_cvpr_2017/html/Chollet_Xception_Deep_Learning_CVPR_2017_paper.html.

43. KOIRALA, A., WALSH, K. B., WANG, Z. and MCCARTHY, C. Deep learning for real-time fruit detection and orchard fruit load estimation: benchmarking of 'MangoYOLO.' **Precision Agriculture**. v. 20, n. 6, p. 1107–1135. 2019. DOI 10.1007/s11119-019-09642-0.

44. BRESILLA, Kushtrim, PERULLI, Giulio Demetrio, BOINI, Alexandra, MORANDI, Brunella, CORELLI GRAPPADELLI, Luca and MANFRINI, Luigi. Single-shot convolution neural networks for real-time fruit detection within the tree**. Frontiers in Plant Science**. v. 10. 2019. DOI 10.3389/fpls.2019.00611.

45. LIU, Z, WU, J, FU, L, MAJEED, Y, FENG, Y, LI, R, ACCESS, Y Cui - IEEE and 2019, undefined. Improved kiwifruit detection using pre-trained VGG16 with RGB and NIR information fusion**. ieeexplore.ieee.org [online]**. Available from: https://ieeexplore.ieee.org/abstract/document/8943411/. Accessed 23 March 2021.

46. MAI, X, ZHANG, H, … X Jia - IEEE Transactions on and 2020, undefined. Faster R-CNN with classifier fusion for automatic detection of small fruits**. ieeexplore.ieee.org [online]**. Available from: https://ieeexplore.ieee.org/abstract/document/8976303/. Accessed 23 March 2021.

47. KANG, H, AGRICULTURE, C Chen - Computers and Electronics in and 2020, undefined. Fast implementation of real-time fruit detection in apple orchards using deep learning**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S0168169919314395. Accessed 23 March 2021.

48. WAN, S, NETWORKS, S Goudos - Computer and 2020, undefined. Faster R-CNN for multi-class fruit detection using a robotic vision system**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S1389128619306978. Accessed 23 March 2021.

49. APOLO-APOLO, OE, MARTÍNEZ-GUANTER, J, … G Egea - European Journal of and 2020, undefined. Deep learning techniques for estimation of the yield and size of citrus fruits using a UAV**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S1161030120300381. Accessed 23 March 2021.

50. MU, Yue, CHEN, Tai-Shen, NINOMIYA, Seishi and GUO, Wei. Intact Detection of Highly Occluded Immature Tomatoes on Plants Using Deep Learning Techniques**. mdpi.com [online]**. DOI 10.3390/s20102984. Available from: www.mdpi.com/journal/sensors. Accessed 23 March 2021.

51. XIONG, J, LIU, Z, CHEN, S, LIU, B, ZHENG, Z, … Z Zhong - Biosystems and 2020, undefined. Visual detection of green mangoes by an unmanned aerial vehicle in orchards based on a deep learning method**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S1537511020300970. Accessed 23 March 2021.

52. VASCONEZ, JP, DELPIANO, J, … S Vougioukas - … and Electronics in and 2020, undefined. Comparison of convolutional neural networks in fruit detection and counting: A comprehensive evaluation**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S016816991932232X. Accessed 23 March 2021.

53. APOLO-APOLO, Orly Enrique, PÉREZ-RUIZ, Manuel, MARTÍNEZ-GUANTER, Jorge and VALENTE, João. A Cloud-Based Environment for Generating Yield Estimation Maps From Apple Orchards Using UAV Imagery and a Deep Learning Technique**. Frontiers in Plant Science**. v. 11. 2020. DOI 10.3389/fpls.2020.01086.

54. FU, L, MAJEED, Y, ZHANG, X, KARKEE, M, ENGINEERING, Q Zhang - Biosystems and 2020, undefined. Faster R–CNN–based apple detection in dense-foliage fruiting-wall trees using RGB and depth features for robotic harvesting**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S1537511020302002. Accessed 23 March 2021.

55. WU, D, LV, S, JIANG, M, AGRICULTURE, H Song - Computers and Electronics in and 2020, undefined. Using channel pruning-based YOLO v4 deep learning algorithm for the real-time and accurate detection of

apple flowers in natural environments**. Elsevier [online]**. Available from: https://www.sciencedirect.com/science/article/pii/S0168169920318986. Accessed 23 March 2021.

56. REN, Shaoqing, HE, Kaiming, GIRSHICK, Ross and SUN, Jian. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks**. IEEE Transactions on Pattern Analysis and Machine Intelligence**. v. 39, n. 6, p. 1137–1149. 2017. DOI 10.1109/TPAMI.2016.2577031.

57. REDMON, Joseph, DIVVALA, Santosh, GIRSHICK, Ross and FARHADI, Ali. *You Only Look Once: Unified, Real-Time Object Detection* [online]. [no date]. Accessed 23 March 2021. Available from: http://pjreddie.com/yolo/.

58. LIU, Wei, ANGUELOV, Dragomir, ERHAN, Dumitru, SZEGEDY, Christian, REED, Scott, FU, Cheng Yang and BERG, Alexander C. SSD: Single shot multibox detector. In : **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**. Springer Verlag, 2016. p. 21–37. ISBN 9783319464473.

59. BOCHKOVSKIY, Alexey, WANG, Chien Yao and LIAO, Hong Yuan Mark. *YOLOv4: Optimal Speed and Accuracy of Object Detection* [online]. 2020. arXiv. Accessed 23 March 2021. Available from: https://github.com/AlexeyAB/darknet.