Fruit Identification and Quality Detection by Means of DAG-CNN

Cesar G. Pachón-Suescún[a,1], Javier O. Pinzón-Arenas[a,2], Robinson Jiménez-Moreno[a,3]

[a] *Department of Mechatronics Engineering, Militar Nueva Granada University, Bogotá D.C, 110111, Colombia*
E-mail: [1]u3900259@unimilitar.edu.co, [2]u3900231@unimilitar.edu.co, [3]robinson.jimenez@unimilitar.edu.co

Abstract— **The design of quality control systems in food has become essential in research to guarantee an adequate state for its consumption. It is necessary to develop automatic and efficient systems that can verify its state before its distribution. This paper presents an algorithm based on deep learning for the identification of fruits and the state they are in, oriented to changes in camera focus, capture angles, lighting variations, and change of backgrounds. In this case, 8 types of fruit are chosen to identify what kind of fruit is being observed and if it is in good condition or not, establishing a total of 16 categories that the network must classify. A convolutional neural network with a DAG structure is proposed for the learning of fruits and their state. A graphic user interface is designed to allow the acquisition of the image of the fruit and its subsequent classification in some of the categories. A 94.43% accuracy was obtained in the 1600 test images classification, with approximate processing times of 45-55 milliseconds. Therefore, it can be concluded that the proposed system based on Deep learning can adequately perform a process of detection of types of fruits and their state.**

Keywords— **convolutional neural network; directed acyclic graph; fruit identification; quality detection.**

I. INTRODUCTION

The design and implementation of quality control systems in industry has covered several areas. A system is designed to detect surface failures on capacitors implementing contour transformation techniques, obtaining a 98.7% accuracy in the detection of defective elements [1]. It is highlighted that the manual detection of failures can cause inaccurate and unreliable information. Convolutional neural networks (CNN) are implemented to detect cracks in infrastructures, obtaining 98.22% in the detection of failures [2]. It was sought to detect the quality of oil plants implementing multi-layer perceptron networks, with entries based on the RGB color scale, its normalized values, a fruit maturity index, and the HSI color scale values, obtaining a 93.5% accuracy in detecting the maturity of the oil plant [3].

On the other hand, the World Health Organization (WHO), highlights the importance of fruit consumption, emphasizing that these can save up to 1.7 million lives each year, besides its low consumption is one of the main factors of mortality in the world [4]. One of the most important factors affecting the consumption of fruits in the population is the state in which they are found. The external appearance might affect consumer behavior [5].

Various investigations have focused on detecting damage to fruits, even prior to harvesting, identifying those that are in an adequate state to be harvested. A light control system is presented to highlight the defects found in fuji apples [6].

The possible candidates containing defects that are seen are classified as spoiled, stem or calyx. Implementing a relevance vector machine from the candidates, the apples are classified as healthy and spoiled, obtaining a 95.63% in the accuracy of the classification of the apples in their respective category. It is sought to detect whether a lemon is fresh or not using a CNN using transfer learning as an identification system and supported by a diffuse system [7]. Hence, it evaluates fruit quality from parameters such as equatorial diameter, surface defects, and fruit weight, obtaining a 97.5% accuracy in lemons classification.

Based on the above and the importance of fruit consumption by people, its study and research become relevant. This is due to additional reasons such as that its superficial state can affect the consumer, where already some investigations have focused on detecting the fruit's state from the characteristics on the surface of these. Most of the current works implement techniques that are not robust to noise, which require the implementation of systems to maintain a constant luminosity and focus on a single type of fruit. In order to make systems based on machine vision more robust, techniques such as CNN have shown their efficiency. For instance, a CNN was implemented to classify 1000 different categories of images among which were included landscapes, foods, and tools [8]. It could obtain 84.7% accuracy in the classification of the images in their respective categories, surpassing the existing results achieved by other types of techniques.

As a result of the CNN boom, they have been implemented in different applications. CNN boom is used for the detection of cancer from magnetic resonance images, obtaining a 98.6% accuracy in the classification of the images [9]. The authors seek to detect vehicles [10], for which a type of CNN is implemented. It is called faster R-CNN, which not only allows identifying the category to which an image but also generates regions in the image where the elements of interest are located without significantly affecting the network classification time [11]. Using this type of network, they put it to the test with the KITTI database [12], evaluating the network in three different types of difficulty, obtaining 95.14%, 83.73%, and 71.22% accuracy in the detection of vehicles. CNNs are implemented to detect and identify specific products in the refrigerator of a home such as milk or juices to alert the user when any of these is not found, obtaining a 96.3% accuracy in identifying the products [13].

In recent years, a variation of the CNN structure has been working, implementing a Directed Acyclic Graph (DAG) structure which is presented in [14]. The main advantage of this type of network is the ability to use several ramifications in the design of the CNN architecture in order to optimize the processing times and learn a greater number of characteristics of the images. An application of a CNN with structure type DAG is presented, focused on the recognition of people´s heartbeat types, where they obtained a 97.15% accuracy in the classification of cardiac abnormalities, surpassing most of the techniques currently used [15].

This article presents as a contribution to the state of the art, evaluation of a DAG-CNN to classify 8 different types of fruit and detect if they are fresh for consumption or not. The first stage corresponds to materials and methods, which presents a general scheme of the implemented system, the database, architecture, and training parameters. The second stage shows the training and results of the network, an analysis of the correct and incorrect classification cases, and the graphic user interface designed to use the network. In the last part of the article, the conclusions about the obtained results are presented.

II. MATERIALS AND METHODS

In Fig. 1, the process to detect the fruit and its state, which consists of three main parts, is shown. The first focuses on the capture of the fruit to be evaluated. Then the image is entered to the DAG-CNN, which is responsible for detecting, from the characteristics of the image, what fruit it is, and its state, where S= Spoiled / F=Fresh.

A. Database FRUIT-16K

For the database's acquisition, it is decided to use 8 different fruits: banana, lemon, lulo, mango, orange, strawberry, tamarillo, and tomato. Two thousand images of each one of the types of fruit are acquired for a total of 16,000, the dataset [16]. Half of them correspond to fresh fruit and the other half to non-fresh fruit. Of the 16,000 images acquired, 10% corresponding to 1600 images were used to validate the network during training, another 10% to test the network after training. Therefore 12800 images corresponding to 80% of the total base were used to train the DAG-CNN.
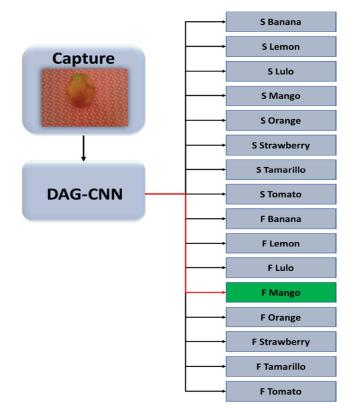

Fig. 1 Scheme of the process of detection of fruit and state

During the database's acquisition, photos were taken to make it robust, changing the backgrounds, rotation, distance of capture, and lighting. The trained network does not require external adjustments for its use. In Fig. 2, some examples of the database for each of the fruits are shown.
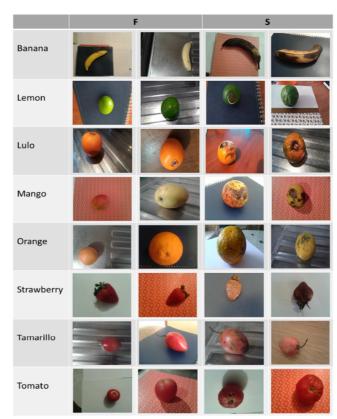

Fig. 2 Samples of the acquired database.

B. Architecture and Training Options

The network architecture was established based on iterative tests, in which variations were made in the number of filters, the organization of the layers, and in the network's depth, obtaining greater learning of the characteristics and precision in the classifications with the architecture presented in Fig. 3.
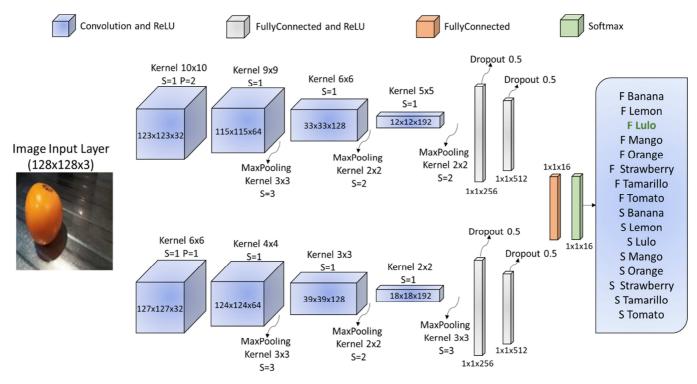

Fig. 3 Architecture of the DAG-CNN where S refers to the Stride and P to the Padding.

A structure type DAG for CNN is established, which consists of two similar branches, where the only difference is the size of the applied filters, this to learn more features in the network. Both branches receive the same image. For this case, the input images are 128×128 pixels; the branch on the left presents filters of a larger size than one at the right, thus ensuring that each will learn characteristics of different sizes details. Each branch consists of 4 convolution layers, 3 down sampling layers (maxpooling), and two fully connected layers. In the CNN, there is a tendency to present a problem called overfitting, this occurs when the network, instead of learning general characteristics, memorizes the images. A study suggested a dropout technique that is in charge of randomly disconnecting a certain percentage of neuron connections in the fully-connected layers during training [17].
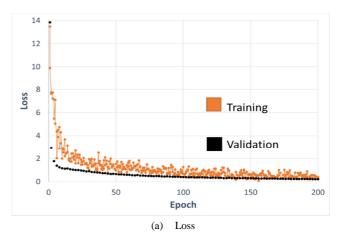
For this work, a dropout of 50% of disconnections is applied. In the last fully-connected layer of the architecture, 16 filters are set. This value corresponds to the number of total categories for which the network was trained. Finally, to know in which category the input image was classified, a Softmax function is used. This is in charge of normalizing the values delivered by the last fully-connected and expressing them in a percentage way. In this way, the category with the highest value will be the one to which the input image belongs.

Based on the training graphs results, it is established that 200 epochs are sufficient for the DAG-CNN to classify the training and validation images accurately. A learning factor of 10-5 is used, in this way, layer weights do not vary aggressively and learning of the network is facilitated. As the total size of the training database is of 12800 images and in order not to overload the GPU, to facilitate the generalization and learning of the characteristics, the network was trained with batch sizes of 32, corresponding each to 0.25% of the total of the training base

C. Training

In Fig. 4, the training of the network in the 200 established epochs is shown. The upper graph shows the network's accuracy with the training and validation images, where 94.25% accuracy was obtained in the classification of the validation dataset during the training.
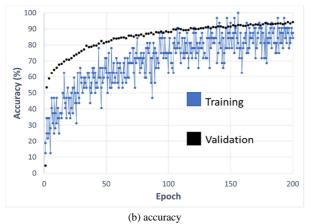

(a)   Loss

2185

(b) accuracy

Fig. 4 Loss and Accuracy in Training of the R-CNN

III. RESULTS AND DISCUSSIONS

After training, the network is tested with 1,600 images other than training and validation, evaluating a total of 100 images per category. Table 1 shows the categories, number of images correctly classified within the category (True Positive), those that were recognized within the evaluated category when they belonged to another (False Positive) and the percentage of accuracy obtained.

TABLE I
CLASSIFICATION IN THE TEST DATASET

Category	True Positive	False Positive	% Precision
S Banana	98	6	98%
S Lemon	98	1	98%
S Lulo	97	2	97%
S Mango	98	5	98%
S Orange	98	3	98%
S Strawberry	96	5	96%
S Tamarillo	82	8	82%
S Tomato	88	13	88%
F Banana	97	3	97%
F Lemon	100	2	100%
F Lulo	99	1	99%
F Mango	93	8	93%
F Orange	98	0	98%
F Strawberry	87	11	87%
F Tamarillo	98	8	98%
F Tomato	84	13	84%
	1511	89	94.43%

The category that presented a greater precision, with 100%, was that of fresh lemons. The category that presented the most errors during the tests was "S Tamarillos," where 4 of them were classified in the "F Tamarillo" category, 5 in "F Tomato" and the rest in other categories. A 94.43% was obtained in the average of the accuracy in the classification of the test images in their respective categories. For this calculation, the true positive and the total of test images in each category were considered. During the network tests, 1600 images were presented, 1511 correctly classified and 89 incorrectly. In Fig. 5, some examples of true and false classifications are presented. In general, false positives were found between the same type of fruit or categories with similar color or shape characteristics.

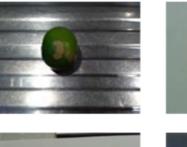True Positive

True Negative

Fig. 5 Examples of network classifications.

In Fig. 6, an example of the CNN activations in one of the fruits in both branches of the network is shown. In each branch, the input image and the most representative activations are presented in the different convolution layers' filters. In general, there are activations of the fruit's network, highlighting that the activation tends to be higher in areas without defects. In the same way, the network can discriminate the fruit's defects, as it is shown in its activations, although there are small areas with slight activations caused by the light reflected on the surface of the peel.
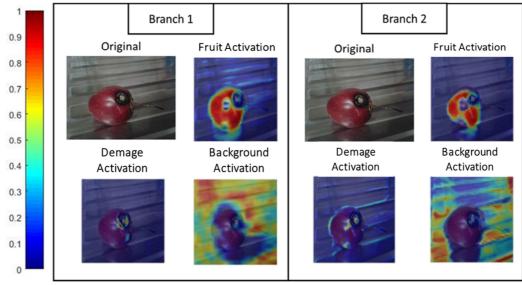
Fig. 6 Activations in both branches of the DAG-CNN.

Finally, an example of background activations is shown, showing that the filters only do not activate in the fruit or defects, they also learn to discriminate the environment from the object of interest. Although in some filters tend to have certain similarities between their activations, especially the most relevant, being the same branches with variations in their kernel, they learn different complimentary details between them. For example, for the activation focused on the damaged area, a branch activates the reflection in one part, while the other does not, but both activate the damaged section, which possibly helps to discriminate in a better way than the fruit effectively if it is spoiled.

Fig. 7 shows the graphic user interface developed, which allows the acquisition of the fruit's image and evidence the classification in any of the categories. In section A, there are two options, where the first one allows to load an image from the database, and the other to take a picture of the fruit, to then display it on the screen. Once the image is loaded or captured, in section B, it can be stored in the database; for doing this, the category that the fruit must previously be selected from the categories list. Also, in this section, when selecting the test option, the image is evaluated by the trained network. Finally, in section C, the category, and the confidence with which the capture was classified are shown.
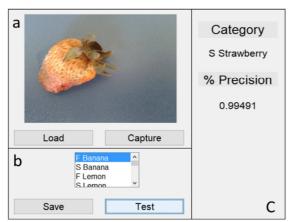

Fig. 7 Graphic user interface.

An important stage of evaluating this type of system is when it takes the program to classify a fruit in its respective category. This depends on its possible implementation in a real environment. Table 2 shows the average times of classification of an image in each of the categories, highlighting that the system was approved in a computer with a sixth-generation core i7 processor, 16 GB RAM DDR4, GPU Nvidia 960M with 4 GB DDR5. Based on the classification times, it is observed that they vary between 45 and 55 ms, reaching a classification of 20 fruits per second approximately.

TABLE II
NETWORK CLASSIFICATION TIMES.

Fruit	Time Fresh (ms)	Time Spoiled (ms)
Banana	43.1	56.2
Lemon	46.9	47.5
Lulo	45.4	46.8
Mango	48.2	48.5
Orange	46.0	46.1
Strawberry	45.2	43.4
Tamarillo	53.2	46.5
Tomato	48.6	53.1

IV. CONCLUSION

The design of a CNN with a DAG structure allows each of the branches to learn characteristics of different sizes, increasing the network's depth without compromising the processing times. Thanks to this, 94.43% of the accuracy of the test images' classification in each of the categories was obtained. It is considering the accuracy obtained with the database by varying backgrounds, position angles, and without implementing systems to control the lighting. It is possible to show that, in comparison to the systems found in state of the art, the proposed system is more robust against disturbances and presents a high degree of accuracy. Based on the processing times, it can be analyzed that the system could be implemented in a machine vision system in real-time, which could work up to 20 fps per second, considering

that the number of frames could be improved by implementing more updated hardware.

For future developments of the system, it is recommended to generate a more detailed analysis of the fruits. It is thought to implement a faster R-CNN to locate the fruit and its defects within the image. It is essential to analyze in more detail if the defect is a disease or physical damage and to identify the problem lies during the cultivation or the harvest and transport.

REFERENCES

[1]   Y. Yang, Z. Zha, M. Gao, and Z. He, "A robust vision inspection system for detecting surface defects of film capacitors," *Signal Processing*, vol. 124, pp. 54-62, 2016. Doi: 10.1016/j.sigpro.2015.10.028.

[2]   Y. Cha, W. Choi, and O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks," *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 5, pp. 361-378, 2017. doi: 10.1111/mice.12263.

[3]   M. Makky, P. Soni, and V. Salokhe, "Automatic non-destructive quality inspection system for oil palm fruits," *International Agrophysics*, vol. 28, no. 3, pp. 319-329, 2014. doi: 10.2478/intag-2014-0022.

[4]   World Health Organization, "Global Strategy on Diet, Physical Activity and Health," [Online]. Available: http://www.who.int/dietphysicalactivity/fruit/en.

[5]   B. Zhang, W. Huang, J. Li, C. Zhao, S. Fan, J. Wu, and C. Liu, "Principles, developments and applications of computer vision for external quality inspection of fruits and vegetables: A review," Food Research International, vol. 62, pp. 326-343, 2014. doi: 10.1016/j.foodres.2014.03.012.

[6]   B. Zhang, W. Huang, L. Gong, J. Li, C. Zhao, C. Liu, and D. Huang, "Computer vision detection of defective apples using automatic lightness correction and weighted RVM classifier," Journal of Food Engineering, vol. 146, pp. 143-151. 2015. doi: 10.1016/j.jfoodeng.2014.08.024.

[7]   C. Enciso-Aragón, C. Pachón-Suescún, and R. Jimenez-Moreno, "Quality control system by means of CNN and fuzzy systems," International Journal of Applied Engineering Research, vol. 13, no. 16, pp. 12846-12853, 2018.

[8]   A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," In Advances in neural information processing systems, 2012, pp. 1097-1105.

[9]   A. Sawant, M. Bhandari, R. Yadav, R., Yele, and M. Bendale, "Brain Cancer Detection From Mri: A Machine Learning Approach (Tensorflow)," Brain, vol. 5, no. 4, pp. 2089-2094, 2018.

[10]   Q. Fan, L. Brown, and J. Smith, "A closer look at Faster R-CNN for vehicle detection," In Intelligent Vehicles Symposium IEEE, 2016, pp. 124-129. doi: 10.1109/IVS.2016.7535375.

[11]   S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," In Advances in neural information processing systems, 2015, pp. 91-99.

[12]   A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," The International Journal of Robotics Research, vol. 32, no. 11, pp. 1231-1237, 2013. doi: 10.1177/0278364913491297.

[13]   C.G. Pachón, J.O. Pinzón, and R.J. Moreno, 2018, "Product Detection System for Home Refrigerators implemented though a Region-based Convolutional Neural Network," International Journal of Applied Engineering Research, vol. 13, no. 12, pp.10381-10388, 2018.

[14]   K. Thulasiraman, and M.N. Swamy, "Graphs: theory and algorithms," John Wiley & Sons, 2011.

[15]   Z. Golrizkhatami, S. Taheri, and A. Acan, "Multi-scale features for heartbeat classification using directed acyclic graph CNN," Applied Artificial Intelligence, vol. 32, no. 7-8, pp. 613-628, 2018. doi: 10.1080/08839514.2018.1501910.

[16]   C.G. Pachón, J.O. Pinzón, and R.J. Moreno, "FRUIT-16K", [Online], Available: https://github.com/DEEP-CGPS/FRUIT-16K.git

[17]   N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," The Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929-1958, 2014.