International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 14, Number 6 (2021), pp. 516-522

Behavior of Three Convolutional Architectures Against Automatic Categorization of the Watermelon Fruit

Heidy M. Rodriguez C.[1], Carolina Mendivelso N.[2] and Fredy H. Martínez S.[3*]

[1]*Facultad Tecnológica, Universidad Distrital Francisco José de Caldas, Colombia.*

[2]*Facultad Tecnológica, Universidad Distrital Francisco José de Caldas, Colombia.*

[3]*Associate Professor, Facultad Tecnológica, Universidad Distrital Francisco José de Caldas, Colombia.*
ORCID: 0000-0002-7258-3909

**Corresponding Author*

Abstract

The Colombian agricultural sector requires the development of technological tools to increase its production and quality levels. In this sense, this research proposes the development of an automatic watermelon fruit classification model for the development of autonomous fruit classification systems. The objective is to determine the best convolutional architecture in terms of performance for the particular problem. Training is performed on three models identified as high performing in this type of problem: ResNet (Residual Neural Network), DenseNet (Dense Convolutional Network), and NasNet (Neural Architecture Search Network). These three architectures are trained equivalently with a proprietary dataset. The results of the three models were evaluated with different metrics both during the training process and during the validation with unknown images to the models. Preliminary results allow choosing the DenseNet topology as the suitable one for the design of an autonomous system.

I. INTRODUCTION

Currently, the implementation of neural networks is of great impact since it has been applied in numerous processes, among the most outstanding ones we find image selection [1], which has been useful to optimize the processes carried out by man to automation systems performed by specialized teams in a required function [2, 3]. Similarly, neural networks have become involved in the field of food products. Taking into account the physical characteristics in the selection of fruits, it is possible to select and establish the most optimal state of the fruit that generates the greatest viability of consumption and profit [4, 5].

Artificial neural networks have been a very useful tool in the area of agriculture since they allow the identification of images of fruits such as watermelons, where an automatic system processes the information of images corresponding to healthy fruits, or in poor condition, optimizing the selection process, offering higher quality in this process through the strategic use of technology [6]. The use of these tools is of great importance when performing food inspection tasks, but to effectively perform this process it is important to train the neural model correctly [7, 8].

Convolutional neural networks have achieved high performance in the industry of categorization of specific elements in images with a very low level of external structuring, achieving the development of highly efficient and very low-cost embedded and autonomous systems [9, 10, 11]. This structure is the one we propose to evaluate to optimize the process of identifying images of agricultural products, intending to increase production and profitability, particularly of watermelon, since the agricultural sector is an important part of the management of economic processes with the greatest impact on the country [12].

The agricultural production of watermelon in Colombia has interesting characteristics that make it stand out internationally. The country has a great variety of thermal floors that facilitate its cultivation. Therefore, and as it has been historically shaped, this fruit is an essential part of the export and economic growth of the sector [13]. The origin of the watermelon, *acendria*, *sindria*, *patilla*, or watermelon is Africa, from where it was imported to Colombia in 1449 [14]. The harvesting characteristics that the country preserves allowed adequate conditions for the cultivation of this product. For its production, a warm and temperate climate is required, with temperatures ranging from 20 to 25 degrees Celsius [15]. According to these characteristics, Colombia has become one of the largest watermelon producers in recent years [16, 17]. Production in 2010 in Colombia reached 100 million tons, representing 11.3 percent of world production, with an annual growth rate of 5.1

percent. By 2016, the selling price of each ton is between 160 and 220 dollars [18].

Based on the benefits obtained from watermelon production, which represent an important part of the country's economy, it is important to consider the different diseases such as fungi, bacteria, and viruses which can be generated by snails, slugs, nematodes, and soil worms. These diseases focus mainly on the youngest watermelons, stopping the growth of the plants, and causing black spots on the watermelon rind. In addition, holes are produced by worms, and a brown and grayish aspect is generated in its decomposition, triggered in the production of these foods, compromising the quality of production. This results in economic losses that affect the national economy [13, 19, 20]. For this reason, the careful selection of watermelon is an important part of the production chain to guarantee the quality of the product. By making the selection through autonomous artificial systems between fruit that is in good condition for consumption and fruit that is diseased or in the process of decomposition, customer protection is guaranteed by offering an export quality product with added value [8, 21].

To perform the procedure of recognizing the state of the fruit, whether it is ripe, green, or in a state of decomposition, image processing must be performed and using fine-tuning the neural models to identify the state in which the fruit is found [22, 23, 24]. This identification was normally done manually, that is, to select the state of the fruit, an operator is located to identify the characteristics of the fruit such as color and size, obtaining information to determine if the fruit is suitable for export [25, 26]. This procedure is intended to be complemented with an artificial system that in the future will eliminate manual inspection, to carry out an automation process that can reduce costs and guarantee quality in the agricultural industry [27, 28].

II. PROBLEM FORMULATION

Watermelon is a fruit of high international export. In Colombia, it is cultivated in several departments such as Huila, Tolima, Meta, Boyacá among others. In addition, it is very healthy since its content is 96 percent water, and contains several vitamins. Considering that watermelon is highly marketed, it is necessary to identify strategies to automate some activities that farmers have, and thus optimize the process of fruit selection, such as determining the appropriate time for harvesting the fruit, and the quality of the harvest. This process is useful in chain stores, where the efficient distribution of fruits can significantly improve the quality of service offered in these stores. And looking at it from another point of view, this sorting procedure can be of great help to people with special needs such as those who lack their visual sense, assisting them in choosing fruit in good condition.

This fruit classification will be done by training a convolutional neural network that processes its layers simulating the visual cortex of the human eye. An own dataset adjusted to the needs of the system to be developed (specific conditions of the classification) must be used. We have a dataset of 300 images divided into three categories: Category 0 with 100 pictures of green fruits, Category 1 with 100 pictures of ripe fruits, and Category 2 with 100 pictures of rotten fruits (Fig. 1). This distribution is on purpose to avoid bias during training. To select the appropriate architecture for fruit categorization, three types of convolutional networks identified as high performing in this type of problem are trained. The selected models implement ResNet, DenseNet, and NasNet networks. Performance evaluation will be performed under similar conditions with the same metrics and both during training and validation with unknown data for each model.

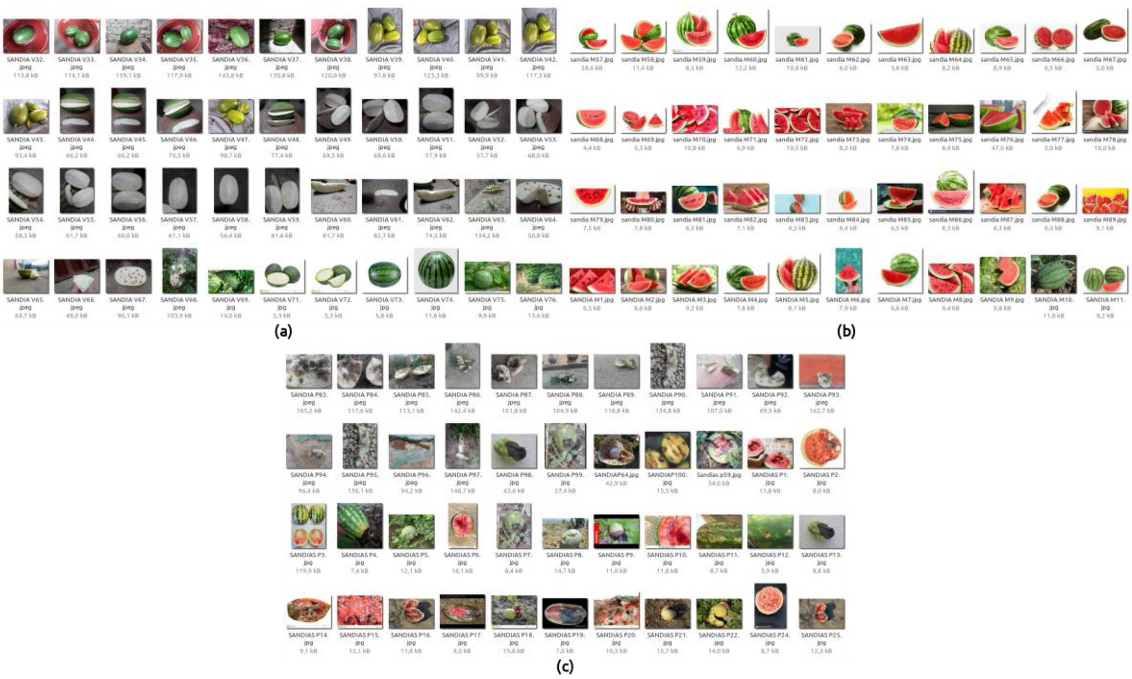

Fig. 1. Sample of the dataset. (a) Category 0 with 100 pictures of green fruits, (b) Category 1 with 100 pictures of ripe fruits, and (c) Category 2 with 100 pictures of rotten fruits

III. METHODOLOGY

For the ResNet (Residual Networks) model we use ResNet50, a reduced convolutional model of the architecture with 50 layers deep, which was trained from scratch (no pre-trained weights). This network is based on forwarding layer hopping (which can be double or even triple) which has been shown to reduce the problem of vanishing gradient and allows forward layers to learn with the same performance as the previous ones. The idea of skipping layers is to omit the problem that arises from vanishing gradients, and thus be able to use the activity generated by previous layers by reusing information from a previous layer. This procedure works best when skipping a single layer that is not linear, or it is also very functional when the intermediate layers are mostly linear.

Dense convolutional neural networks (DenseNet) are a variation, or rather an improvement of residual neural networks. They correspond to a densely connected convolutional network architecture or dense networks since it is necessary to improve the learning of the neural network by increasing the number of layers of the network. In these networks, each of the layers of the neural network has as input the characterization maps of the previously analyzed layers, and in turn, the layers that have already been characterized are used as inputs in all the following layers. In these dense blocks, each layer takes all previous feature maps as input, which helps the training process by relieving the leakage gradient problem. This leakage gradient problem appears in really deep networks whereby backpropagating the error in the network, this error is reduced at each step and eventually becomes 0. These connections allow the error to propagate further without reducing too much. Our model used the DenseNet121 architecture with 121 layers deep and trained from scratch.

The Neural Architecture Search Network (NasNet) seeks to identify blocks with high performance on specific datasets. This architecture adapts the architecture of other neural networks (such as those mentioned above) to build small blocks with small datasets, and then to transfer this block to a larger dataset. We use the NasNetMobile architecture that has been optimized for the ImageNet database, but as in the previous cases, it was trained from scratch.

In all three cases, the same data set was used, separated in the same percentages between training (70%) and validation (20%), but with different images in each case since they were always randomly mixed. The Categorical Cross Entropy function was always used as loss and the Stochastic Gradient Descent (SGD) as optimizer. During training, Accuracy and Mean Squared Error metrics were calculated at each epoch for both training and validation data. Precision, Recall, and F1-Score metrics were also calculated with the validation data for each of the three final models. This information together with the confusion matrices and the ROC (Receiver Operating Characteristic) curves served as a reference for the performance evaluation of the models.

IV. RESULTS AND DISCUSSION

The training of the ResNet model was performed for 30 epochs, it was terminated when it was observed that the model had learned the training data, and little by little its performance with the validation data was poorer (Fig. 2). At epoch 17 the Accuracy of the model for the training data was 97%, and for the validation data 61%, however, this was its best performance point. Thereafter the Accuracy for new data during validation dropped steadily to less than 20%. The neural network was overfitted, and even at its best point, its performance for unknown data was quite poor.

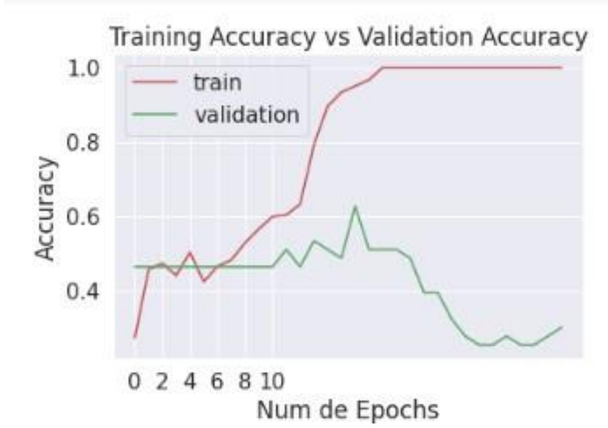

Fig. 2. Model behavior based on training and validation data (Accuracy - ResNet)

Similar behavior is observed in Fig. 3. This figure shows the behavior of the losses during the same 30 epochs, also for training and validation data. Again, it is observed that the error drops to zero for the training data at the same instant that Accuracy increases, but for the validation data the value stagnates and tends to grow.

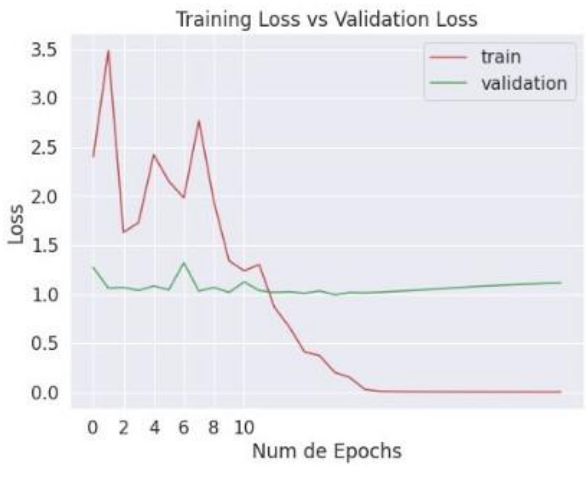

Fig. 3. Model behavior based on training and validation data (Loss - ResNet)

The confusion matrix (Fig. 4) and the Precision, Recall, and F1-Score metrics (Fig. 5) for this model were calculated for the validation data (images unknown to the model). The results confirm what was observed during training; the model miscategorizes images into the second category, resulting in an average Precision of less than 20%, an average Recall of 35%, and an average F1-Score of 24%, with values of zero in all three cases for the second category (Category 1).

International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 14, Number 6 (2021), pp. 516-522

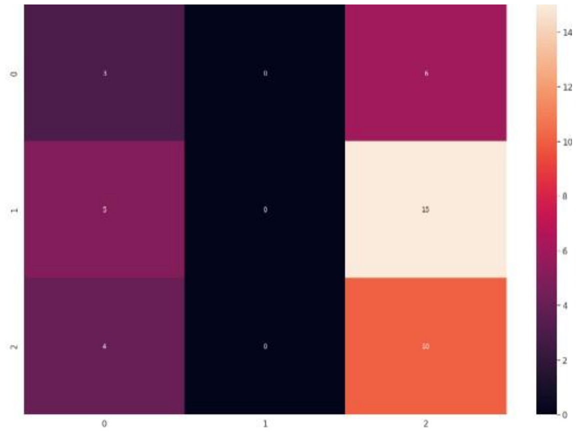

Fig. 4. Confusion matrix (ResNet)

	precision	recall	f1-score	support
0	0.25	0.33	0.29	9
1	0.00	0.00	0.00	20
2	0.32	0.71	0.44	14
accuracy			0.30	43
macro avg	0.19	0.35	0.24	43
weighted avg	0.16	0.30	0.20	43

Fig. 5. Model metrics (ResNet)

The DenseNet model performs better overall concerning the validation data. Fig. 6 shows the Accuracy behavior during training for both training and validation data. The training of this model was performed for 55 epochs, at which point the overall performance was considered to have reached its optimal behavior. Throughout the training, the Accuracy of the training data increased, but in the last 15 epochs, this growth stalled at 55%. The data corresponding to validation grew during the first 30 epochs and then stalled at around 40%. Although this final value is low, it exceeds that achieved by the ResNet model.

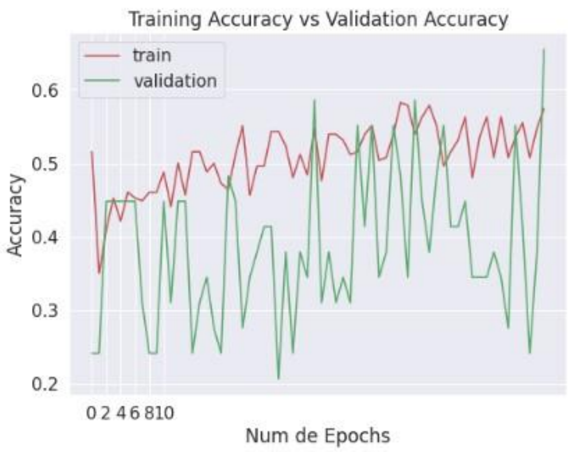

Fig. 6. Model behavior based on training and validation data (Accuracy - DenseNet)

The error at each epoch for training and validation data is shown by the loss function in Fig. 7. During the first few epochs, the losses drop below 10%, remaining small for the rest of the training. This suggests that it is difficult to improve the performance of this model.

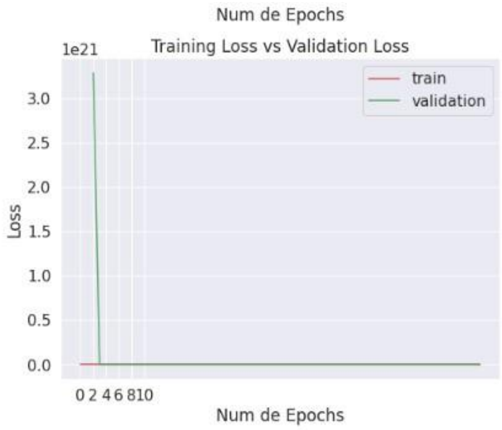

Fig. 7. Model behavior based on training and validation data (Loss - DenseNet)

Although the DenseNet model showed better performance for the validation data than the ResNet model, the confusion matrix (Fig. 8) and the metrics applied to the model with this unknown data (Fig. 9) show poor behavior in terms of classifying images within the correct classification (as measured by the Recall parameter), with only a small performance improvement. The confusion matrix shows the same classification problems, with most of the images being misclassified in the second category (Category 1), and missing most of the images in the first category (Category 0). The metrics report some improvement over the ResNet model, the average accuracy rises to 76% (a large increase over the previous model which only achieved 19%), but this is achieved by placing most of the images in Category 0, even though there are images that do not fall into this category. This is observed by having an accuracy of 100% for Category 0, but only 65% for the other two categories. In addition, the average Recall was 59%, a metric that weights only the correct classifications (the Recall for Category 0 was only 14%), and the average F1-Score was 56% (weighting of Precision and Recall).

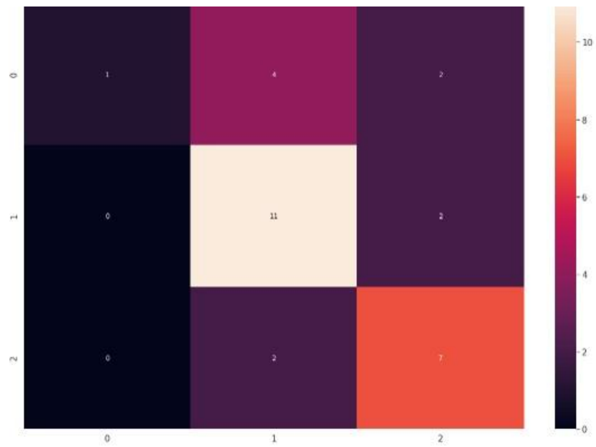

Fig. 8. Confusion matrix (DenseNet)

	precision	recall	f1-score	support
0	1.00	0.14	0.25	7
1	0.65	0.85	0.73	13
2	0.64	0.78	0.70	9
accuracy			0.66	29
macro avg	0.76	0.59	0.56	29
weighted avg	0.73	0.66	0.61	29

Fig. 9. Model metrics (DenseNet)

Finally, the NasNet model was trained for 50 epochs under the same parameters and optimizers used in the previous two models. Fig. 10 shows the behavior of the accuracy during this training, both for training and validation data. Similar to the previous two models, the error for the training data reduces very quickly (after 10 epochs), and remains small for the rest of the training. During most of the training, the accuracy for the validation data remained at 38%, but after epoch 40 this value reached 48%. The training was stopped to avoid overfitting.

Fig. 10. Model behavior based on training and validation data (Accuracy - NasNet)

Fig. 10 shows similar behavior for the loss function. The value drops steadily over the 50 epochs for the validation data (a fact that was not very evident in Fig. 10) but remains constant and low for the training data after epoch 20.

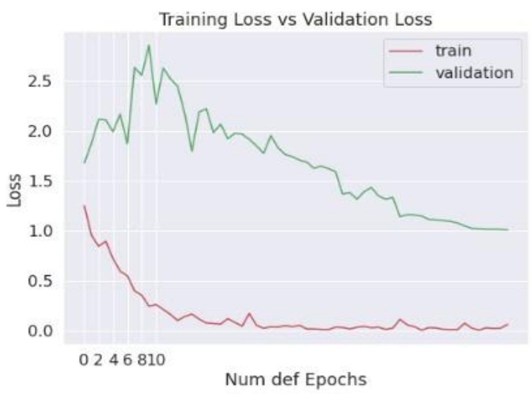

Fig. 11. Model behavior based on training and validation data (Loss - NasNet)

The confusion matrix (Fig. 12) again shows model-specific problems regarding the correct categorization of the validation images. Most of the images are placed in Category 1, which makes many match this category, and the accuracy of the model increases, but categories 0 and 1 reduce considerably their number of correctly classified images. This is observed again in Fig. 13, the accuracy for Category 2 reached 83%, but the average only achieves 53% due to the low performance of Category 0. The Recall value in Category 1 reached 94%, indicating that the images placed there effectively belong to the category, but in the other two categories, only 15% was achieved, for an average Recall of only 41%. The same behavior was observed in the F1-Score value, which only achieved an average value of 34%.

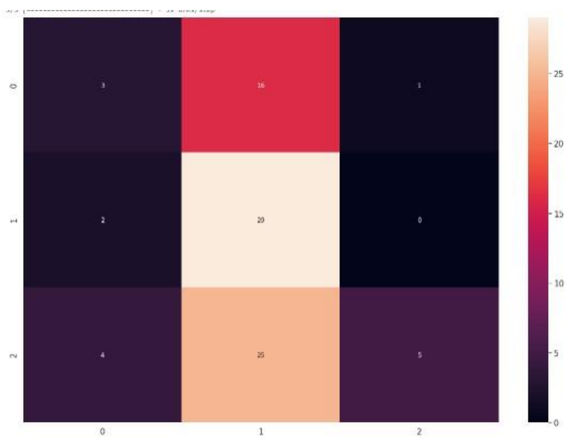

Fig. 12. Confusion matrix (NasNet)

	precision	recall	f1-score	support
0	0.33	0.15	0.21	20
1	0.41	0.94	0.57	31
2	0.83	0.15	0.25	34
accuracy			0.44	85
macro avg	0.53	0.41	0.34	85
weighted avg	0.56	0.44	0.36	85

Fig. 13. Model metrics (NasNet)

When comparing the performance of these three models against our dataset, the DenseNet architecture not only achieved the highest values in the metrics, but it is the only model that according to the trends in the training process can improve its performance by fine-tuning the model. With an increase in the size and variability of the dataset, as well as adjustments in the optimization and loss functions and the use of pre-trained parameters, it is possible to increase the performance of this model, and take it to the development of an autonomous embedded system for automatic fruit sorting.

V. CONCLUSION

This paper studies the behavior of three convolutional architectures for the automatic categorization of the watermelon fruit. The three selected models correspond to topologies previously identified by the research group as high performing in similar tasks. The topologies used were: ResNet (Residual Neural Network), DenseNet (Dense Convolutional Network), and NasNet (Neural Architecture Search Network). These three models were trained with the same dataset, which consists of 300 images separated into three categories according to the state of the fruit. For training, this dataset was randomly separated into a training group (70%) and a validation group (the remaining 30%). Although the number of epochs was adjusted to the behavior of each model, the three models were trained with similar conditions, i.e., they all used the same loss function (Categorical Cross Entropy function) and the same optimizer (Stochastic Gradient Descent), the training was performed from scratch without the use of pre-trained parameters, and the algorithms were not manipulated to increase their performance with the validation data. To assess the performance of the models, different metrics were calculated both during and after training. During training, accuracy and loss were calculated at each epoch for both validation and training data. The trend of these two functions is fundamental to estimate the learning and generalization capability of each model. After training, the Precision, Recall, and F1-Score metrics were used to evaluate the performance of each model against new data (images from the validation group). The confusion matrix of each model with these new images was also calculated. After analyzing the results, it was observed that the best performing model was the DenseNet architecture, which not only showed the best numbers both individually per category and on average, but according to the curves it even has learning and generalization capabilities. This research continues with a fine-tuning of this model which includes both tunings of the dataset and the convolutional model itself.

ACKNOWLEDGMENTS

This work was supported by the Universidad Distrital Francisco José de Caldas, specifically by the Facultad Tecnológica. The views expressed in this paper are not necessarily endorsed by Universidad Distrital. The authors thank all the students and researchers of the research group ARMOS for their support in the development of this work.

REFERENCES

[1] C. J. Tablada and G. A. Torres, “Redes neuronales artificiales”, Revista de educación matemática, vol. 24, no. 3, 2009.

[2] X. B. Olabe, “Redes neuronales artificiales y sus aplicaciones”, Publicaciones de la Escuela de Ingenieros, 1998.

[3] F. Martínez, C. Penagos, and L. Pacheco, “Scheme for motion estimation based on adaptive fuzzy neural network”, Telkomnika, vol. 18, no. 2, pp. 1030–1037, 2020, ISSN 1693-6930, doi:10.12928/TELKOMNIKA .v18i2.14752.

[4] G. Figueredo and J. Ballesteros, “Fruit ripeness identification with artificial neural networks - a review”, Ciencia y Agricultura, vol. 13, no. 1, pp. 117–132, Jan 2016, doi:10.19053/01228420.4811.

[5] C. J. Carrion Osnayo, “Aplicación de redes neuronales artificiales para la predicción de la recuperación de planta concentradora en minsur sa–unidad san rafael”, 2018.

[6] D. E. Figueroa, E. R. Guerrero, et al., “Sistema de visión artificial para la identificación del estado de madurez de frutas (granadilla)”, Redes de Ingeniería, vol. 7, no. 1, pp. 78–86, 2016.

[7] P. N. Constante Prócel and A. M. Gordón Garcés, Diseño e implementación de un sistema de visión artificial para clasificación de al menos tres tipos de frutas, Master’s thesis, Quito, 2015., 2015.

[8] E. da Silva, I. da Silva Carmo, J. Monteiro Neto, R. de Medeiros, M. Augusti, J. Amaya, H. Murga-Orrillo, and C. Abanto-Rodríguez, “Cultivation of nine varieties of watermelon under edaphoclimatic conditions of the brazilian savannah: Morphological variables, physicochemical characteristics and useful life of fruits [cultivo de nueve variedades de sandía bajo condiciones edafoclimáticas de la sabana brasilera: Variables morfológicas, características fisicoquímicas y vida útil de frutos]”, Scientia Agropecuaria, vol. 11, no. 4, pp. 493–501, 2020, doi:10.17268/SCI.AGROPECU.2020.04.04.

[9] E. del Agua and S. D. C. E. de Camarón, “Modelo basado en redes neuronales artificiales”.

[10] H. Montiel, E. Jacinto, and F. Martínez, “Recognition of fissures in bony structures through image processing”, International Journal of Engineering and Technology, vol. 10, no. 4, pp. 1223–1229, 2018, ISSN 2227-524X.

[11] F. Martínez, C. Hernandez, and A. Rendón, “Identifier of human emotions based on convolutional neural network for assistant robot”, Telkomnika, vol. 18, no. 3, pp. 1499–1504, 2020, ISSN 1693-6930.

[12] A. L. Feltrim, A. B. CecÃlio Filho, M. V. Gonsalves, L. C. Pavani, J. A. C. Barbosa, and J. W. M. Cortez, “Distancia entre plantas y dosis de nitrÃy potasio en sandÃa sin semillas fertirrigada ”, Pesquisa AgropecuÃ¡ria Brasileira, vol. 46, pp. 985 – 991, 09 2011, ISSN 0100-204X.

[13] P. Ferreñan and J. Enrique, “Sistema de visión artificial para apoyar en la identificación de plagas y enfermedades del cultivo de sandía en el distrito de ferreñafe”, 2019.

[14] A. Romero, A. Marín, and J. A. Jiménez, “Sistema de clasificación por visión artificial de mangos tipo tommy”, Revista UIS Ingenierías, vol. 14, no. 1, pp. 21–31, 2015.

[15] W. Barreto and R. Picón, “Experimental and simulation study of compose panels inelastic behavior using artificial neural networks [estudio experimental y simulación del comportamiento inelástico de paneles compuestos usando redes neuronales artificiales]”, Informes de la Construccion, vol. 72, no. 558, pp. 1–10, 2020, doi:10.3989/ic.70957.

[16] J. Zhang, H. Huang, Z. Xia, L. Ma, Y. Duan, Y. Feng, and J. Huang, "Artificial neural networks for chemistry representation in numerical simulation of the flamelet-based models for turbulent combustion", IEEE Access, vol. 8, pp. 80020–80029, 2020, doi:10.1109/ACCESS.2020.2990943.

[17] M. P. Machuca-Infante, O. E. Vélez-Santamaria, and Y. A. Heredia-Gómez, "Oportunidad comercial de exportar sandía colombiana al mercado italiano", 2021.

[18] G. A. A. Mogrovejo, H. D. R. Ibarra, F. U. de Popayán, and A. de Empresas Agropecuarias, "Análisis de los indicadores técnico económicos del cultivo de sandía citrullus lanatus en el municipio de patía departamento del cauca".

[19] G. A. F. Avila, "Clasificación de la manzana royal gala usando visión artificial y redes neuronales artificiales.", Res. Comput. Sci., vol. 114, pp. 23–32, 2016.

[20] C. F. Espinal, H. J. Martínez Covaleda, and Y. Peña Marín, "La cadena de los frutales de exportación en colombia: Una mirada global de su estructura y dinámica 1991-2005", 2005.

[21] M. Mazarakis, J. Custer, M. Kiefer, J. Leckbee, D. Anderson, G. Raymond, T. Tung, F. Wilkins, and R. Obregon, "High-power dielectric diode studies at sandia national laboratory", IEEE Transactions on Plasma Science, vol. 48, no. 11, pp. 4048–4055, 2020, doi:10.1109/TPS.2020.3028748.

[22] L. Rascón-Madrigal, M. Sinecio-Sidrian, J. Mejía-Muñoz, J. Díaz-Román, I. Canales-Valdiviezo, and A. Botello-Arredondo, "Estimation of hand-grip intention: Cylindrical, spherical and hook using deep neural networks [estimación en la intención de agarres: Cilíndrico, esférico y gancho utilizando redes neuronales profundas]", Revista Mexicana de Ingenieria Biomedica, vol. 41, no. 1, pp. 117–127, 2020, doi:10.17488/RMIB.41.1.9.

[23] R. Cordero-Chan, M. G. Orozco-del Castillo, M. R. Moreno-Sabido, J. J. Hernández-Gómez, G. Cetzal-Balam, and C. Couder-Castaneda, "Clasificación de galaxias utilizando procesamiento digital de imágenes y redes neuronales artificiales", Research in Computing Science, vol. 147, no. 7, pp. 319–332, 2018.

[24] A. Barrero, M. Robayo, and E. Jacinto, "Board navigation algorithm in controlled environments based on image processing", Tekhnê, vol. 12, no. 2, pp. 23–34, 2015, ISSN 1692-8407.

[25] E. M. T. Ocampo, D. A. M. GIRALDO, and H. S. Isaza, "Pronóstico de ventas usando redes neuronales", Scientia et technica, vol. 10, no. 26, pp. 25–30, 2004.

[26] J. Diaz and M. Perez, "Identificación de formas geométricas y colores mediante procesamiento y reconocimiento de imágenes", Tekhnê, vol. 2, no. 1, pp. 9–17, 2004, ISSN 1692-8407.

[27] M. J. Andrade-Cuvi, C. Moreno-Guerrero, M. Guijarro-Fuertes, and A. Concellón, "Caracterización de la naranjilla (solanum quitoense) común en tres estados de madurez", Revista Iberoamericana de Tecnología Postcosecha, vol. 16, no. 2, pp. 215–221, 2015.

[28] A. Pitarque, J. F. Roy, and J. C. Ruiz, "Redes neurales vs modelos estadísticos: Simulaciones sobre tareas de predicción y clasificación", Psicológica, vol. 19, pp. 387–400, 1998.