Acta Scientiarum
http://periodicos.uem.br/ojs
ISSN on-line: 1807-8664
Doi: 10.4025/actascitechnol.v47i1.71113

COMPUTER SCIENCE

Development of an artificial vision algorithm with neural networks to detect coffee berry borer in coffee beans

Ryan Abraham León León*, Martin Antonio Renteria Davila, Bryan Alexander Lara Guerra, Anapaula Manosalva Ruiz and Cristina Azucena Sotomayor Montenegro.

Facultad de Ingeniería y Arquitectura, Universidad Cesar Vallejo, Perú. *Author for correspondence. E-mail: rleonle@ucvvirtual.edu.pe

ABSTRACT. This research addresses the problem of coffee berry borer in coffee plantations, which causes significant economic losses to companies in the coffee industry. This pest affects the quality of the coffee beans, reducing their market value and decreasing revenues. To address this issue, the use of artificial vision with neural networks is proposed as a solution. The main objective of this study is to develop a computer vision algorithm using convolutional neural networks to detect coffee bean defects caused by the coffee berry borer through image processing. Ultimately, this research aims to implement a coffee bean classification solution that achieves an efficiency of 91.8% based on the given criteria.

Keywords: Algorithm; coffee berry borer; computer vision; cost-effective.

Received on January 24, 2024.
Accepted on June 20, 2024.

Introduction

Currently, many companies involved in the production and marketing of coffee are affected by the presence of the coffee berry borer. This pest can cause significant economic losses as the affected beans are considered to be of very low quality and sold at low prices. According to Mansilla et al. (2021), coffee is one of the most widely consumed beverages in the world because of its taste, aroma, and the alertness it induces in people. It is derived from the beans of the coffee plant. In Peru, coffee is the main agricultural export, ranking seventh in the world. There are 425,416 hectares of coffee plantations in Peru, with the potential to expand to two million hectares. As noted by Cargua et al. (2022), the coffee market employs approximately two million Peruvians, representing one third of agricultural employment.

According to Herrera et al. (2016b), coffee quality is currently defined by both intrinsic and extrinsic properties, including physical aspects such as size, humidity, heat, and possible defects in the beans. Coffee cultivation requires precise environmental conditions and faces challenges from pests, particularly the coffee berry borer (CBB). As noted by Rosas et al. (2019), this pest can damage plants, affect their development, and reduce the quality of the beans and the final product. Tovar et al. (2022), state the coffee berry borer is the most aggressive pest in global coffee cultivation, causing significant losses both in Central African countries where it is endemic and in regions where it has been introduced.

According to Quintero et al. (2019), the CBB causes a significant loss of 5% of annual coffee production in many countries, equivalent to 300 thousand quintals and 120 million soles. Portocarrero et al. (2023), report the pest has been found in several regions of the country. In addition, a CBB infestation of 24% can reduce the yield by 16.3% and the grain weight by 21.2 grams per 100 grams. To control CBB, growers use techniques such as pruning, removal of affected fruit, use of pesticides, intercropping, and natural pesticides. However, Araújo et al. (2023), stress that the use of pesticides has negative environmental and health effects. Some coffee farmers use a traditional method called the "float method" or "siphon tank method" to sort coffee berries. According to Vásquez et al. (2022), this method requires high water consumption (about 4.7 L kg-1g), and the water cannot be reused due to contamination.

Today, the implementation of artificial vision has facilitated the detection of quality defects in coffee beans and plants. This methodology combines optical and computational techniques to assess the presence of the CBB in beans. As described by Araújo et al. (2019), images captured with an optical lens are analyzed using the Python programming language. In one study, a neural network algorithm was developed to classify coffee beans by color and detect the presence of CBB. The algorithm achieved 97% effectiveness in recognizing the ripening stage, demonstrating the viability of artificial vision as a quality control method.

In addition, automatic diagnosis of CBB-infected threshed coffee beans was proposed. Fifty images of beans, both with and without CBB, were acquired to train computer vision algorithms using support vector machine modeling techniques. Teodoro et al. (2015), found that the algorithm achieved an optimal classification accuracy of 90%, demonstrating its effectiveness in diagnosing CBB infection.

On the other hand, JM (2017) proposes the use of machine vision as a low-cost solution for sorting coffee beans instead of relying on expensive traditional equipment. Machine vision sorting takes less time (1 hour) compared to manual sorting (2 hours). In addition, machine vision has lower costs and is more flexible in terms of quality parameters. In conclusion, machine vision sorting of coffee beans is superior and less expensive than traditional methods.

As a result, Herrera et al. (2016a), found machine vision is crucial for identifying coffee beans affected by CBB during classification, which helps to increase coffee farmers' income. An algorithm was developed that uses coffee bean segmentation and a pixel mask to identify CBB characteristics, allowing the identification of affected beans with an accuracy of about 90%.

The main objective of the present study is to develop an artificial vision algorithm using neural networks to detect defects in coffee beans caused by CBB through image processing. The first specific objective is to use the necessary system libraries for the proper development and execution of the algorithm. The second specific objective is to design and implement a structure that provides suitable LED illumination to facilitate the accurate detection of coffee beans in a specific environment.

Metodology

In the present study, image processing algorithms were used in the PyCharm 2023.1.2 software, an IDE used to create and analyze programs, allowing the identification of errors in real time. In addition, the PIP tool, a preferred installation program that allows the installation, reinstallation, or uninstallation of packages with a single command, was used to perform the processing. The OpenCV library was also used, which contains more than 2500 algorithms focused on computer vision and machine learning.

For image processing, a system of illumination and conditioning was designed to obtain images of the coffee beans (Figure 1). The setup included 24W low power LED lights with a white background. The LED lights were strategically placed at the top to provide uniform illumination and avoid unwanted reflections in the images of the coffee beans, resulting in clear and accurate images.

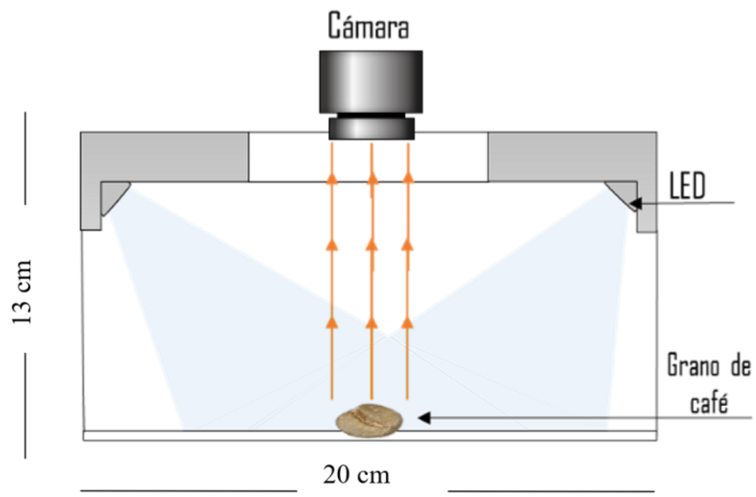

Figure 1. Image capture structure.

In Figure 1, the images were taken at approximately 13 cm to ensure adequate focus and detection of the coffee beans. A camera with a 12 MP ultrawide angle lens and an F2.2 aperture was used.

Figure 2 shows both healthy and unhealthy coffee beans were captured to evaluate the differences in characteristics and to teach the algorithm how to detect imperfections. A total of 120 images of coffee beans were used, of which 100 were unhealthy and 20 were healthy, ensuring that the program could be properly trained to detect the imperfections.

Figure 2. Healthy and unhealthy coffee beans.

According to Vives et al. (2014), for feature extraction, the 'pip install labelme' function was used to install the 'labelme tool'. This tool allows for labeling through points in an image, which allows for accurate identification of objects and imperfections. At this stage, image segmentation is performed, which is a fundamental step in image processing. In this process, the image is divided into smaller parts based on color in order to select the most relevant sections.

As shown in Figure 3, imperfections were identified in one of the images, and the same process was applied to the other images by converting them to JSON format. The 'labelme2yolo' function was then installed to convert the files to the YOLO format, an algorithm that uses neural networks to detect objects in images in real time. The Yolov8m version was used for this work. Aparicio et al., (2023), recommend using the 'labelme2yolo --json_dir' command, which was used to run the previous function and extract the training and validation folders. Next, the ultralytics library was installed using the 'pip install ultralytics' function, which helped to train and evaluate detection, segmentation, and classification models. Finally, the following command was used (see Figure 4):

Figure 3. Feature identification.

```
command = "yolo task=segment mode=train epochs=20 data=dataset.yaml model=yolov8m-seg.pt

imgsz=640 batch=2"
```

Figure 4. Training command.

Command shown in Figure 4 is used to start training the YOLO model to segment the CBB in the coffee beans.

Once the training of the images is finished, we proceed to execute the algorithm shown in Table 1:

Table 1. Algorithm functions for detection.

FUNTION	DESCRIPTION
model= YOLO (“best.pt”)	Contains the previously trained weights of the model.
cap = cv2.VideoCapture (2)	Used to initiate real-time video capture from the camera.
ret, frame = cap.read ()	Used to read frame from the camera, "red" if the reading was successful, "frame" contains the captured frame.
results = model.predict(frame, imgsz = 640, conf = 0.3)	Used to perform the detection using the YOLO model and the results obtained must have that image size and the confidence threshold value equal to or greater than 0.3.
marks = results [0].plot ( )	Used to display the detected objects or features and their labels.
cv2.imshow(“BBC detection”, marks)	Shows the image with the "CBB detection" annotations using OpenCV.
cv2.waitKey(1) = = 27:	It is used to verify if the "ESC" key has been pressed, which in ASCII code is the number 27, it is used to stop programming and will wait for 1 millisecond after pressing the key.
cap.release( )	It is used to free the camera, so that it is available for other programs.
cv2.destroyAllWindows( )	It is used to close all the windows opened by OpenCV, helping to free resources when they are no longer needed in the display windows.

To detect the characteristics in the coffee beans, logistic regression will be applied to predict the coordinates of the bounding boxes and the probabilities of the types of characteristics detected in an image (Table 2).

Table 2. Function — equation.

FUNCTION	EQUATION 1
result = model.predict(frame, imgsz = 640, conf=0.3)	$P(y = 1\|x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}}$

In addition, a confidence threshold was used to adequately filter the beans for the detection of CBB in coffee beans. If the detection confidence is greater than or equal to 30, it is considered a valid detection (Table 3).

Tabla 1. Function — equation.

FUNCTION	EQUATION 2
conf=0.3	$Confidencia \geq umbral$

Finally, to process the coffee bean images, a basic technique called kernel filtering was used to highlight the characteristics and eliminate noise in the images (Table 4).

Tabla 2. Function – equation

FUNCTION	EQUATION 3
cv2.imshow("BBC Detection", marks)	$g(x, y) = \sum_{i=k}^{k} \sum_{j=-k}^{k} f(x - i, y - j) h(i; j)$

The dark shades in the images were identified as CBB, facilitated by the use of a white background. For real-time identification of CBB, the Iriun Webcam application was used, which focused on the lighting system and displayed the results of the algorithm execution on the laptop.

As shown in Figure 5, the first step is to convert the images to grayscale to reduce noise and facilitate the filtering process. Next, blurring is applied to obtain the maximum number of features of coffee beans affected by CBB. Red circles indicate the detection of corners with CBB in the beans, followed by thresholding using the Canny method. In the next step, the features are read to identify the imperfections of the beans in real time, and then the features are merged with the detected matches. For more accurate detection, ‘labelme’ was used to identify CBB by delineating the infected areas with polygons and labeling the parts marked as infected. Finally, all the images were trained to obtain their features, resulting in the correct detection and labeling of the infected coffee beans.

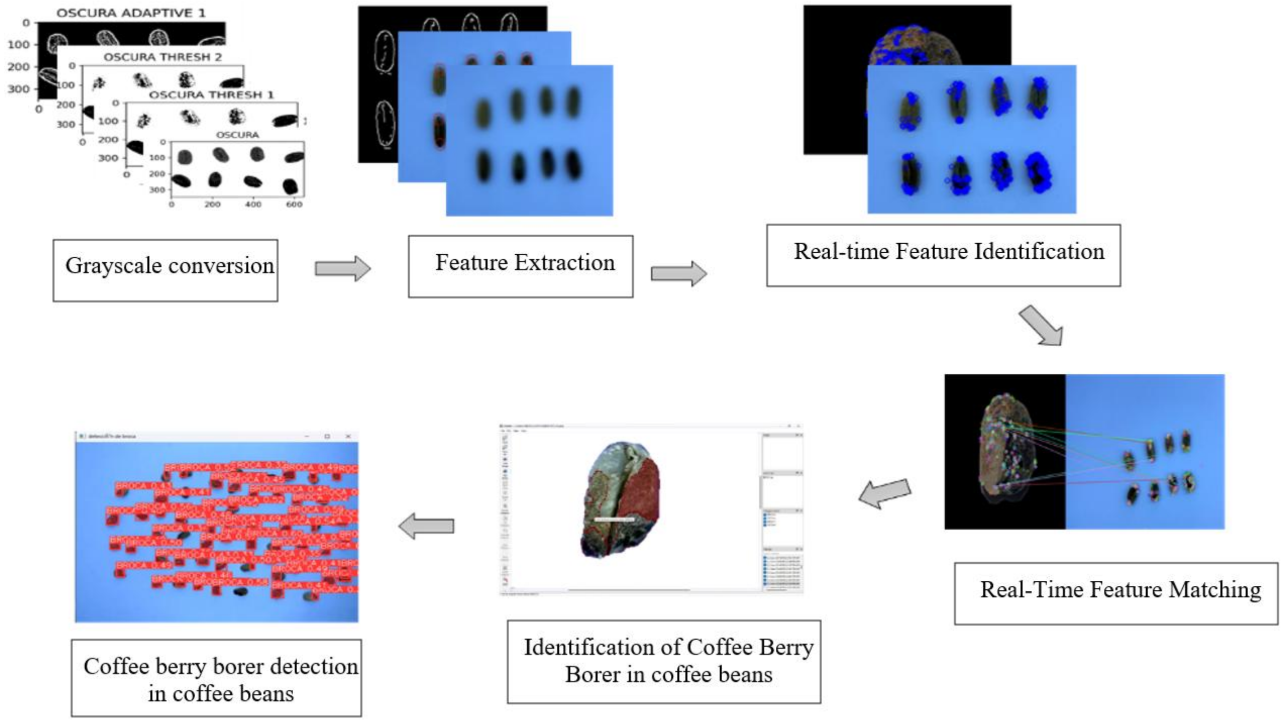

Figure 5. Algorithm development.

Figure 6 illustrates the acquisition of images of both healthy and unhealthy coffee beans using a camera with a 12 MP ultrawide-angle lens and a focal length of 2.2. The process continues with the processing of these captured images, where feature extraction, identification, and real-time detection are performed. This is followed by the execution of the artificial vision algorithm using neural networks. This algorithm detects unhealthy coffee beans by displaying a message with red lines highlighting the imperfections. If the algorithm detects a healthy coffee bean, no message is displayed.

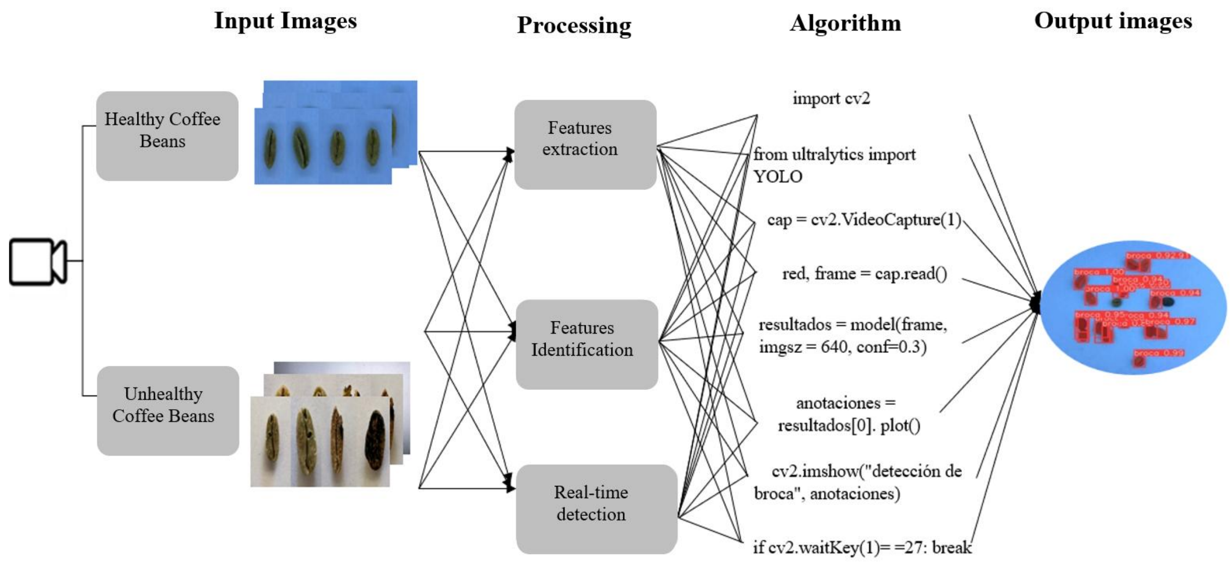

Figure 6. Artificial vision architecture.

Results

The results are based on the identification of unhealthy coffee beans using the developed detection system. For this purpose, a Table 5 has been created containing the analyzed samples of coffee beans. The unhealthy beans that were correctly detected are counted, and with this data the overall efficiency is calculated. This efficiency is represented by the proportion of beans detected by CBB relative to the total sample. (60 samples).

To calculate the efficiency of the algorithm after analyzing photographs of multiple coffee beans, the following equation is used (Equation 4):

$$E = \frac{N_d}{N} \times 100 \quad (4)$$

E = Efficiency.
N_d = Number of unhealthy coffee beans identified.
N = Total number of grains analyzed.

Table 3. Tests performed.

,	Detected	Non-Detected	Efficiency
1	55	5	91.7%
2	55	7	88.3%
3	55	4	93.3%
4	56	6	90.0%
5	55	5	91.7%
6	55	5	91.7%
7	57	3	95.0%
8	53	5	91.7%
9	56	5	91.7%
10	54	4	93.3%
Mean			91.8%

Figure 7 shows the percentage efficiency of each test, showing that test #7 has the highest percentage at 95%. However, the average efficiency across all tests was calculated to determine the overall efficiency. The segmentation algorithm was able to detect unhealthy coffee beans with an overall efficiency of 91.8%.

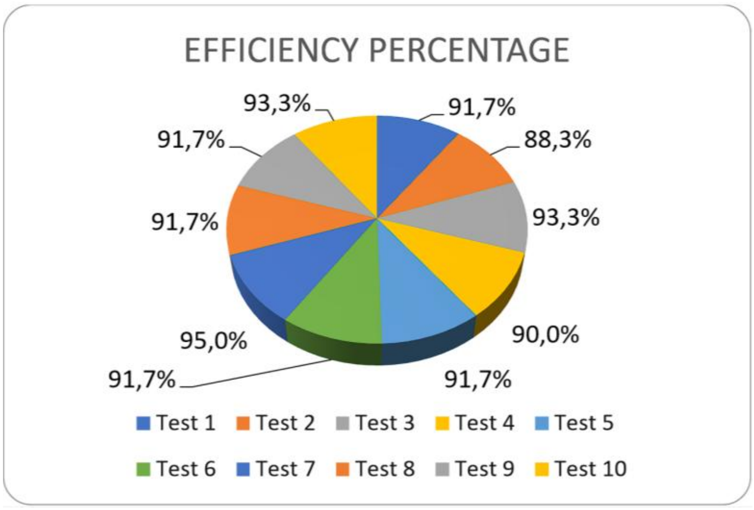

Figure 7. Percentage efficiency in each test.

Discussion

In this work, Castrillón et al. (2017), extended the scope of the algorithm to classify healthy and various defective coffee beans, especially those with CBB. For classification, a perforation index was introduced by dividing the total number of pixels of each bean by the detected dark pixels. Specific true positives for CBB, healthy and defective beans were obtained, with the overall efficiency of the classifier set at 90%. Tools such as Scilab and C++, along with the OpenCV library, were used to analyze the images. Grain classification was performed by observing RGB intensities and analyzing morphologies in an environment with a controlled lighting system to maintain moisture in the grains during inspection. A CanonScan® 5600F scanner and an Emergent® Vision Technologies HS 2000C camera were used for image acquisition.

On the other hand, in our research, an identification system was implemented and a table of analyzed samples of coffee beans was created. The efficiency of the algorithm is calculated using an equation that considers the proportion of correctly detected beans in relation to the total sample. The tests performed show an average efficiency of 91.8% in the detection of CBB. The system is implemented using PyCharm with PIP as the preferred installation tool, and the OpenCV library is used for image analysis, with a focus on computer vision and machine learning. This setup is used to extract features and identify defects in coffee beans in a controlled lighting environment.

Compared to the research by Castrillón et al. (2017), our research stands out for its specific focus on the detection of CBB, providing solid and consistent efficiency in this aspect. It shows a significant improvement in the selection and use of tools and technologies for image processing. The use of PyCharm and PIP for package management reflects a more modern and efficient approach. Regarding image acquisition, our research highlights the quality of the camera used, with a 12 MP ultrawide angle lens and a 2.2 MP focal aperture, which is a significant improvement over the scanner and camera mentioned in the research by Castrillón et al. (2017). In addition, our controlled illumination system with energy-efficient LED lights avoids unwanted reflections and ensures uniform illumination. In terms of image analysis, our research uses updated tools such as PyCharm, PIP, and OpenCV, highlighting their versatility and robustness.

Conclusion

The algorithm for the detection of CBB in coffee beans using neural networks was correctly developed, achieving an overall efficiency of 91.8% in the ten tests performed, as shown in the results table.

Artificial vision techniques were applied through PyCharm, including feature extraction, edge detection, and finally the detection of the CBB. Libraries such as OpenCV, Ultralytics, and Yolov8m were used to ensure proper development and execution of the algorithm. These libraries were essential for detecting and extracting the specific characteristics of each coffee bean used as a sample in the development of the algorithm.

Finally, a suitable LED illumination structure was implemented, which improved the accuracy of CBB detection in coffee beans. This resulted in greater efficiency and quality in the classification and detection process.

References

Aparicio, L., Amaya, O., & Devia, P. (2023). Aplicación de deep learning para la identificación de defectos superficiales utilizados en control de calidad de manufactura y producción industrial: Una revisión de la literatura. *Ingeniería*, *28*(1). https://doi.org/10.14483/23448393.18934

Araújo, C., Oliveira, L., & Eça, G. (2023). Counting of shoots of *Eucalyptus sp.* clones with convolutional neural network. *Pesquisa Agropecuária Brasileira*, *58*. https://doi.org/10.1590/S1678-3921.pab2023.v58.03363

Araújo, C., Souza, P., Assis, A., Cabacinha, C., Leite, H., Soares, C., Silva, A., & Castro, R. (2019). Artificial neural networks, quantile regression, and linear regression for site index prediction in the presence of outliers. *Pesquisa Agropecuária Brasileira*, *54*, e00078. https://doi.org/10.1590/S1678-3921.pab2019.v54.00078

Cargua, J., Luna, A., González, H., Cedeño, G., & Cedeño, A. (2022). Crecimiento y calidad de plantas de café arábica con la aplicación de biochar y biofertilizantes en vivero. *Chilean Journal of Agricultural & Animal Sciences*, *38*(1), 3–14. https://dx.doi.org/10.29393/chjaas38-lccja50001

Castrillón, J., Sanz, J., & Ramos, P. (2017). Algoritmo para la identificación de café lavado afectado por la broca del café. *Revista Cenicafé*, *68*(2), 7–19. https://biblioteca.cenicafe.org/bitstream/10778/1081/1/arc068%2802%297-19.pdf

Herrera, J., Medina, S., Beleño, K., & Gualdrón, O. (2016a). Diseño de un sistema automático de selección de frutos de café mediante técnicas de visión artificial. *Revista UIS Ingenierías*, *15*(1), 7–14. https://doi.org/10.18273/revuin.v15n1-2016001

Herrera, J., Medina, S., Martinez, G., Beleño, K., & Berrio, J. (2016b). Clasificación de los frutos de café según su estado de maduración y detección de la broca mediante técnicas de procesamiento de imágenes. *Scientia Agropecuaria*, *14*(1), 15–22. http://www.scielo.org.co/scielo.php?pid=S1692-82612016000100002&script=sci_arttext

Mansilla, R., Espejo, R., Bernacchia, G., Wither, J., Quispe, C., & López, C. (2021). Genetic diversity and population structure of a Peruvian *Coffea arabica* L. collection. *Chilean Journal of Agricultural Research*, *81*(2), 138–150. https://dx.doi.org/10.4067/S0718-58392021000200138

Portocarrero, A., Vilca, E., Ortiz, B., Miranda, L., & Jiménez, H. (2023). Adaptación de inteligencia artificial por el modelo de regresión múltiple estocástica para determinar la calidad de la fibra de alpaca (*Lama pacos*). *Revista de Investigaciones Veterinarias del Perú*, *34*(2). https://dx.doi.org/10.15381/rivep.v34i2.23130

Quintero, A., Ríos, L., Quintana, D., & León, B. (2019). Sistema experto para el diagnóstico presuntivo de enfermedades fúngicas en los cultivos. *Revista Cubana de Ciencias Informáticas*, *13*(1), 61–75. http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2227-18992019000100061&lng=es&tlng=es

Rosas, C., Solís, H., & Cueva, A. (2019). Sistema eficiente y de bajo costo para la selección de granos de café: Una aplicación de la visión artificial. *Scientia Agropecuaria*, *10*(3), 347–351. https://dx.doi.org/10.17268/sci.agropecu.2019.03.04

Teodoro, P., Barroso, L., Nascimento, M., Torres, F., Sagrilo, E., Santos, A., & Ribeiro, L. (2015). Redes neurais artificiais para identificar genótipos de feijão-caupi semiprostrado com alta adaptabilidade e estabilidade fenotípicas. *Pesquisa Agropecuária Brasileira*, *50*(11), 1054–1060. https://doi.org/10.1590/S0100-204X2015001100008

Tovar, T., Calvo, F., & Bejarano, A. (2022). Desarrollo de un sistema de clasificación de imágenes digitales para medir la humedad en granos de café. *Información Tecnológica*, *33*(3), 117–128. https://dx.doi.org/10.4067/S0718-07642022000300117

Vásquez, W., Inga, M., & Betalleluz, I. (2022). Inteligencia artificial en acuicultura: Fundamentos, aplicaciones y perspectivas futuras. *Scientia Agropecuaria*, *13*(1), 79–96. https://dx.doi.org/10.17268/sci.agropecu.2022.008

Vives, L., Mejía, H., Vilcherrez, K., & Vassallo, M. (2014). Visión artificial: Aplicación de filtros y segmentación en imágenes de hojas de café. *Ingeniería: Ciencia, Tecnología e Innovación*, *1*(2), 71. https://doi.org/10.26495/icti.v1i2.198