Computer Vision-Based Foot Measurement Using Modified Canny Edge Algorithm

Ronald Rance Allorde
School of Electrical, Electronics and Computer Engineering
Mapua University
Manila, Philippines
rrcallorde@mymail.mapua.edu.ph

Carlo Fornoles
School of Electrical, Electronics and Computer Engineering
Mapua University
Manila, Philippines
cpfornoles@mymail.mapua.edu.ph

Meo Vincent Caya
School of Electrical, Electronics and Computer Engineering
Mapua University
Manila, Philippines
mvccaya@mapua.edu.ph

2023 IEEE 5th Eurasia Conference on IOT, Communication and Engineering (ECICE) | 979-8-3503-1469-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/ECICE59523.2023.10383174

***Abstract*—Algorithms of the Canny Edge Detection have been used to highlight the edges of an object in an image. It has been used to determine the sizes of things by applying machine visual or reference things. In this study, we developed an image capture device to obtain an image of the foot by implementing an image processing algorithm to measure the foot size. We evaluated the proposed system by measuring foot sizes using a t-test. We used the modified canny edge algorithm using the foot size measuring system with a median filter. Additionally, we determined the different factors that hindered the accuracy of the system. The factors included the contrast between the foot and the background, the deviation of the foot from the center of the platform, and the thresholds used in the different parts of the canny edge algorithm. The difference between the actual and the system-measured foot size was found to be insignificant. The computed p-value, 0.1809, was less than the critical value, 0.05, and the mean absolute error, 0.3729cm, was less than the margin of equivalence, 0.5cm.**

Keywords—modified canny edge algorithm, median filtering, image processing, foot size measuring system, edge detection

I. INTRODUCTION

Foot size is significant in picking the right shoe for the comfort and health of the wearer. Wearing the wrong shoe size causes swelling and deformities and develops severe complications. Designing shoes that fit is costly as the time and effort are required to measure the contour of the sole user's foot [1]. In addition, foot sizes are used in medical practices and other commercial products [2]. The edges of target objects in previous studies are used in many applications [3]. Edge detection has been used for object detection and identification in computer vision [4–9]. Nevertheless, the canny edge algorithm is the most commonly used technique [10–17]. When measuring the object's size, a few problems arise due to the distance between the camera and the thing. Many techniques have been developed to solve these problems, using a reference object within the same image. The reference object was available such as a coin, with its actual size recorded. The ratio between the reference object's exact size and image size is used to calculate the measurement of the main object [2].

On the other hand, image filtering removes noise and other unwanted parts of images such as the contours of different objects, which is helpful in computer vision object detection [18]. The pre-processing part of the canny edge algorithm is the filter application and is proven to be an essential part of the technique [11]. It is significant in the system's accuracy as it clears out unnecessary parts of the image and prevents the loss of edge details of objects in the image [13]. The Median Filter is one of the commonly used filters as a part of the object detection process [10–13,15,18]. Noise needs to be removed while preserving the necessary contrast, which is essential information about the image [10].

The study's main objective was to calculate the foot size of a person using image processing. We developed an image capture device to obtain a foot image by implementing an image processing algorithm to process the image and measure the foot size. The performance of the proposed system was compared with a standard method of measuring foot size using a t-test.

II. METHODOLOGY

The following tools and equipment were used to develop the prototype for foot size measurements. The prototype was created the necessary environment to obtain a clear image of the foot contour with high contrast between the background and the foot itself.

A. Raspberry Pi 4 Model B

We used a Raspberry Pi 4, specifically a model B, to run the Python script containing the image filtering and processing techniques necessary to generate an accurate measurement. The Python script was auto-started at boot using the crontab. It was placed beside the Okdo 5mp Camera as the CSI camera connector was short, as shown in Fig. 1.

B. OKdo 5mp Camera

The OKdo 5mp camera is a Raspberry Pi OS-compatible camera used to capture images of the foot. It was directly connected to the Raspberry Pi via a CSI camera connection. The camera was placed directly at the bottom center of the foot platform shown in Fig. 1.

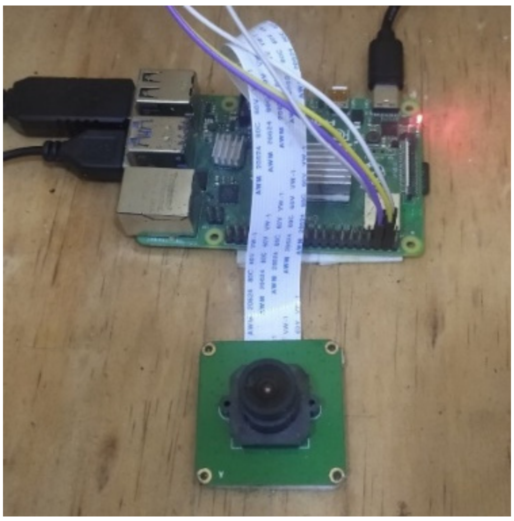

Fig. 1. Raspberry Pi connection to OKdo 5MP camera.

C. Frame and Materials

The prototype's frame was made of a plyboard with dimensions 11.38 x 14.1 x 32 in as shown in Fig. 2. The plyboard was used to make the prototype sturdy enough to support some of the user's weight when placing their foot on the platform. The LCD showed the resulting measurement computed by the system in cm. It was connected to the Raspberry Pi using jumper wires and placed at the top of the frame for the user to read the resulting measurement easily. The Glass was used as the platform of the foot to lessen noise in the image and was placed 15 in above the bottom of the frame for the camera to capture the entire platform.

Fig. 2. Prototype frame made from plyboard.

Two lights generated a more precise image with high contrast between the foot and the background. Each was placed on both sides of the frame just above the platform as shown in Fig. 3. Both were connected to the Raspberry Pi through the USB port, as shown in Fig. 1. Furthermore, a button was added to allow the user to scan when the foot rested in the platform's center. The switch was placed in an optimal position that was the easiest to reach for all users.

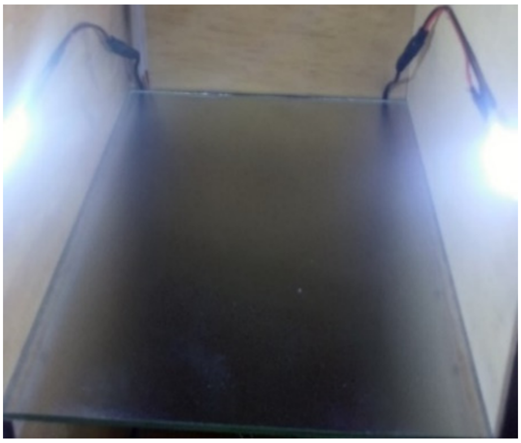

Fig. 3. Translucent glass platform with USB-powered lights.

III. Methodology

This study was conducted in two major parts: prototype development and data-gathering.

A. Development of Prototype

The proposed system's conceptual framework is shown in Fig. 4. The image of the user's foot was taken as input. The image went through a series of processes: Edge Detection, Hysteresis Thresholding, and Bounding Box Computation. Afterward, the Hysteresis thresholding was isolated from the foot contour from the image before applying the bounding box. Lastly, the system's final output was the user's Foot Size.

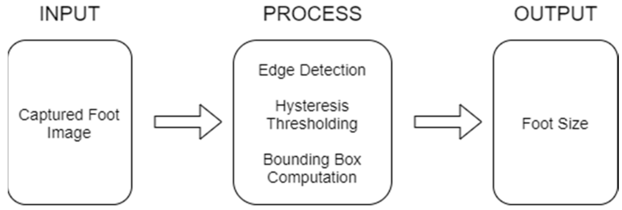

Fig. 4. Conceptual framework.

The prototype software was created based on the original Canny Edge Algorithm with alterations for improved system accuracy. The Canny Edge Algorithm is a multi-stage image processing technique that outputs a clear edge or contour of the objects in the image with a low error rate [19–23]. The Canny Edge Algorithm, partnered with the generation of bounding boxes can be used for detection [24]. It also allows systems to detect features of objects for identification [25]. However, it also measures the object's size. The process starts with obtaining the foot image. Other image processing-related studies used a pre-processing method on the images for enhanced performance [26–30]. Instead of using the common filtering technique in the Canny Edge Algorithm and Gaussian Filtering, the Median Filtering Technique was used as it preserved the contrast between objects in an image. Afterward, the bounding box of the foot contour was computed and applied with a linear regression model to calculate the ratio between the pixels and the actual measurement. The foot size was displayed on the LCD as the system's output as shown in Fig. 5.

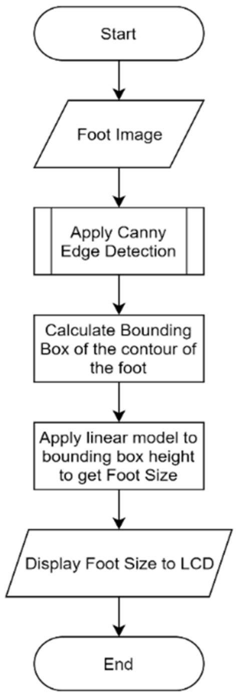

Fig. 5. Algorithm flowchart.

The software was created in 7 Python scripts, including preparing and measuring the user's foot. In other studies, OpenCV was used with image processing packages to build models to perform faster than other packages [25,28,29,31].

The preparation was conducted to generate csv.py and linear.py. _csv.py was generated in the CSV file of the samples. linear.py was created in the CSV file to create the linear model for the pixel-to-centimeter ratio. A Python package called sklearn to generate the linear regression model [32].

The evaluate.py was run for the sample foot images to show the actual and generated measurement of the system and determine the system's accuracy. The index.py was used as the main Python script of the measuring system and originated from the functions of the foot_measurement.py and canny_edge_detection.py to generate the foot size measurement. It contained the code to activate the camera and take a picture of the foot with a 10-second delay for the user to place their foot on the center of the platform.

The canny_edge_detection.py used the entire canny edge algorithm and a Median filter instead of the traditional Gaussian filter for enhanced performance to preserve the contrast between the objects and blur the unnecessary details within the image [12,13]. It was followed by the typical process of the canny edge algorithm to reach the hysteresis thresholding. The thresholds were selected through trial and error with the best results on the sample foot image. Figure 6 shows the foot image before and after applying the canny edge algorithm.

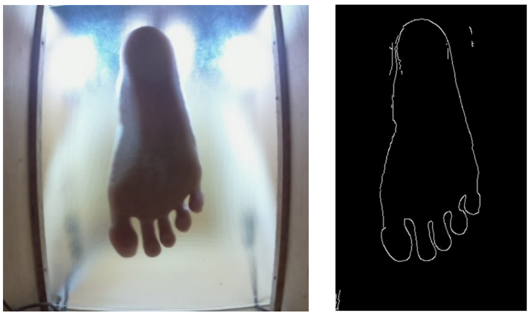

Fig. 6. Before (left) and after (right) applying canny edge algorithm.

The config.py was configured for image-slicing to remove the edges of the sides of the platform. The isolation allowed the edges of the foot to be the most visible in the image and further increased its accuracy. Figure 7 shows the difference between before and after the picture was sliced.

Fig. 7. Before (left) and after (right) slicing image.

B. Data Gathering

The following hypotheses were proposed based on the results.

H$_O$: The means of the real measurement and the measurement calculated by the system are not equal.

H$_A$: The means of the real measurement and the measurement calculated by the system are equal within a margin of equivalence (delta).

A total of 30-foot images using the prototype were taken. Each participant had their right foot size recorded using a measuring tape for the actual measurement data. Half of the data was used to train a linear model, and the other half was used for testing. The real and generated measurement was then placed in a table.

IV. Results

The data gathered from the 30 participants are shown in Table I. The first column shows the actual foot size measurement of the participants using the measuring tape, while the second column shows the actual measurement generated by the system.

TABLE I. FOOT SIZE DATA

No.	Actual Measurement	Computed Measurement
1	27.00cm	27.14cm
2	24.10cm	23.50cm
3	26.50cm	26.60cm
4	23.10cm	23.40cm
5	24.50cm	24.10cm
6	23.10cm	23.30cm
7	25.40cm	24.89cm
8	23.20cm	24.59cm
9	15.50cm	14.24cm
10	19.30cm	18.46cm
11	21.30cm	21.91cm
12	20.50cm	20.18cm
13	25.40cm	25.49cm
14	24.20cm	24.17cm
15	18.40cm	19.70cm
16	21.30cm	21.08cm
17	22.20cm	21.47cm
18	18.50cm	18.57cm
19	23.20cm	23.14cm
20	16.50cm	16.09cm
21	24.60cm	24.61cm
22	22.00cm	21.66cm
23	24.00cm	23.68cm
24	23.80cm	24.22cm
25	23.70cm	24.30cm
26	23.70cm	23.23cm
27	24.10cm	23.29cm
28	23.60cm	23.18cm
29	23.50cm	24.06cm
30	21.30cm	21.28cm

The data was processed with paired T-test using Eqs. (1), (2), and (3). The d value of the actual and generated measurement samples using (1). In the equations, n refers to the number of total samples, and i refers to the computed current sample. The d-values were then used to calculate the standard deviation of the samples, as shown in (2).

$$d = \frac{\sum_{i=1}^{n} d_i}{n} \quad (1)$$

The computed values were then used to calculate the t-value with a p-value of 0.05.

$$s^2 = \frac{\sum_{i=1}^{n}[d_i - \bar{d}_i]^2}{n} \quad (2)$$

The mean absolute error was compared to a set value, delta = 0.5 cm, to see if there was a discrepancy between the actual and calculated values of the foot length.

$$t = \frac{d}{s/\sqrt{n}} \quad (3)$$

The resulting t-value was 1.4080, which was s equivalent to a P-value of 0.1809, greater than its critical value of 0.05. The average error was 0.3729 cm, smaller than the set delta of 0.50 cm.

V. Conclusion and Recommendations

Since the calculated p-value was larger than the critical value of 0.05, the difference between the actual foot size and the computer-generated foot size measured by the proposed system was insignificant. The average error of 0.3729 cm was smaller than the set delta of 0.50 cm, indicating that there was a discrepancy between the two measurements, but within the accepted margin of error. There are factors affecting the accuracy of the system. Foot edges were lost in the canny edge algorithm due to the dark-colored objects such as dark pants as the shadow of the foot lost its high contrast contour image. Another factor was the placement of the foot. The more the foot deviated from the center of the platform, the more it looked smaller from the camera's point of view. The median filter preserved edges enough to have an accurate bounding box around the foot. Lastly, the camera produced lower-quality pictures to readjust and focus. A 10-second interval after the camera activation was needed for a more precise and high-contrast image. It is necessary to use other regression models, and pre-processing filtering techniques and refine the thresholding processes.

Acknowledgment

We want to extend our sincerest gratitude to our school, Mapúa University, for providing us with the education and training that enabled us to complete this paper. Our Program Chair, Sir Noel Linsangan, Ma'am Analyn Yumang, Dr. Jocelyn Villaverde, and our adviser, Sir Meo Vincent Caya, serve special recognition for their guidance. Their valuable insights and direction were instrumental in making this study a success. We would also like to extend our gratitude to our families and friends who participated in this study and offered their support. Without their assistance, including financial support from our parents, we would not have been able to carry out this research.

References

[1] M. F. Mohd Hamzah, O. Mohd Rijal, K. Kimura, and N. Mohd Noor, "Malaysian Women Shoe Sizing System Using Multivariate Normal Probability Distribution," IEEE Access, vol. 7, pp. 142726–142737, 2019, doi: 10.1109/ACCESS.2019.2944763.

[2] M. A. Shahid, M. A. Iftikhar, Z. A. Gondal, M. Adnan, and S. Rathore, "Object Size Measurement through Images: An Application to Measuring Human Foot Size," in *2018 International Conference on Frontiers of Information Technology (FIT)*, IEEE, Dec. 2018, pp. 298–302. doi: 10.1109/FIT.2018.00059.

[3] E. J. L. Aguilar, G. K. P. Borromeo, and J. Flores Villaverde, "Determination of Pineapple Ripeness Using Support Vector Machine for Philippine Standards," in *2021 IEEE 7th International Conference on Control Science and Systems Engineering (ICCSSE)*, IEEE, Jul. 2021, pp. 283–287. doi: 10.1109/ICCSSE52761.2021.9545163.

[4] D. J. M. Bonifacio, A. M. I. E. Pascual, M. V. C. Caya, and J. C. Fausto, "Determination of Common Maize (Zea mays) Disease Detection using Gray-Level Segmentation and Edge-Detection Technique," in *2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, IEEE, Dec. 2020, pp. 1–6. doi: 10.1109/HNICEM51456.2020.9399998.

[5] R. Dela Cruz, S. Guevarra, and J. F. Villaverde, "Leaf Classification of Costus Plant Species Using Convolutional Neural Network," in *2023 15th International Conference on Computer and Automation Engineering (ICCAE)*, IEEE, Mar. 2023, pp. 309–313. doi: 10.1109/ICCAE56788.2023.10111389.

[6] A. Yumang, G. Magwili, S. K. C. Montoya, and C. J. G. Zaldarriaga, "Determination of Shelled Corn Damages using Colored Image Edge Detection with Convolutional Neural Network," in *2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, IEEE, Dec. 2020, pp. 1–6. doi: 10.1109/HNICEM51456.2020.9400023.

[7] C. D. D. Yu and J. F. Villaverde, "Avocado Ripeness Classification Using Graph Neural Network," in *2022 14th International Conference on Computer and Automation Engineering (ICCAE)*, IEEE, Mar. 2022, pp. 74–79. doi: 10.1109/ICCAE55086.2022.9762435.

[8] B. J. Andujar, N. J. Ferranco, and J. F. Villaverde, "Recognition of Feline Epidermal Disease using Raspberry-Pi based Gray Level Co-occurrence Matrix and Support Vector Machine," in *2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, IEEE, Nov. 2021, pp. 1–6. doi: 10.1109/HNICEM54116.2021.9731971.

[9] D. A. Padilla, J. F. Villaverde, J. J. T. Magdaraog, A. J. L. Oconer, and J. P. Ranjo, "Vehicle and Weather Detection Using Real Time Image Processing Using Optical Flow and Color Histogram," in *2019 5th International Conference on Control, Automation and Robotics (ICCAR)*, IEEE, Apr. 2019, pp. 880–883. doi: 10.1109/ICCAR.2019.8813346.

[10] X. Zhang, S. Bao, C. Yang, S. Mad, and J. Yang, "An External Contour Extraction Method for Pollen Images," in *2021 2nd International Conference on Artificial Intelligence and Computer Engineering (ICAICE)*, IEEE, Nov. 2021, pp. 691–695. doi: 10.1109/ICAICE54393.2021.00135.

[11] X. Zhang and H. Yuan, "Edge detection of printed matter based on improved Canny operator," in *2022 IEEE 10th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, IEEE, Jun. 2022, pp. 2546–2549. doi: 10.1109/ITAIC54216.2022.9836680.

[12] D. Saranyaraj, "Image De-noising and Edge Segmentation using Bilateral Filtering and Gabor-cut for Edge Representation of a Breast Tumor," in *2022 International Conference on Engineering and Emerging Technologies (ICEET)*, IEEE, Oct. 2022, pp. 1–6. doi: 10.1109/ICEET56468.2022.10007228.

[13] S. Wang, X. Liao, and G. Wu, "Infrared Image Edge Detection Based on Improved Canny Algorithm," in *2021 IEEE 3rd Eurasia Conference on IOT, Communication and Engineering (ECICE)*, IEEE, Oct. 2021, pp. 280–284. doi: 10.1109/ECICE52819.2021.9645606.

[14] A. Pandey and S. K. Shrivastava, "A Survey Paper on Calcaneus Bone Tumor Detection Using different Improved Canny Edge Detector," in *2018 IEEE International Conference on System, Computation, Automation and Networking (ICSCA)*, IEEE, Jul. 2018, pp. 1–5. doi: 10.1109/ICSCAN.2018.8541194.

[15] A. S. Mohammed Shariff, R. Bhatia, R. Kuma, and S. Jha, "Vehicle Number Plate Detection Using Python and Open CV," in *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, IEEE, Mar. 2021, pp. 525–529. doi: 10.1109/ICACITE51222.2021.9404556.

[16] Handoyo *et al.*, "Gray Scale and Edge Detecting Method To Extract Raw Data in The Diffusivity Measurement System," in *2018 3rd International Seminar on Sensors, Instrumentation, Measurement and Metrology (ISSIMM)*, IEEE, Dec. 2018, pp. 58–60. doi: 10.1109/ISSIMM.2018.8727642.

[17] N. A. OTHMAN, M. U. SALUR, M. KARAKOSE, and I. AYDIN, "An Embedded Real-Time Object Detection and Measurement of its Size," in *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, IEEE, Sep. 2018, pp. 1–4. doi: 10.1109/IDAP.2018.8620812.

[18] A. A. Kumar, N. Lal, and R. N. Kumar, "A Comparative Study of Various Filtering Techniques," in *2021 5th International Conference on Trends in Electronics and Informatics (ICOEI)*, IEEE, Jun. 2021, pp. 26–31. doi: 10.1109/ICOEI51242.2021.9453068.

[19] C. D. Cabalar, S. Marie Cleofas, and M. V. Caya, "Identification of Philippine Chili Peppers using Canny Edge Detection and Feed – Forward Artificial Neural Network," in *2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, IEEE, Dec. 2022, pp. 1–6. doi: 10.1109/HNICEM57413.2022.10109560.

[20] M. V. Caya, D. Padilla, G. Ombay, and A. J. Hernandez, "Detection and Counting of Red Blood Cells in Human Urine using Canny Edge Detection and Circle Hough Transform Algorithms," in *2019 IEEE 11th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, IEEE, Nov. 2019, pp. 1–5. doi: 10.1109/HNICEM48295.2019.9072708.

[21] J. K. Medina, P. Jasper P. Tribiana, and J. F. Villaverde, "Disease Classification of Oranda Goldfish Using YOLO Object Detection Algorithm," in *2023 15th International Conference on Computer and Automation Engineering (ICCAE)*, IEEE, Mar. 2023, pp. 249–254. doi: 10.1109/ICCAE56788.2023.10111494.

[22] J. N. Uy and J. F. Villaverde, "A Durian Variety Identifier Using Canny Edge and CNN," in *2021 IEEE 7th International Conference on Control Science and Systems Engineering (ICCSSE)*, IEEE, Jul. 2021, pp. 293–297. doi: 10.1109/ICCSSE52761.2021.9545195.

[23] M. R. Abalos and J. F. Villaverde, "Fresh Fish Classification Using HOG Feature Extraction and SVM," in *2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, IEEE, Dec. 2022, pp. 1–5. doi: 10.1109/HNICEM57413.2022.10109442.

[24] M. V. Caya, E. D. Arturo, and C. Q. Bautista, "Dog Identification System Using Nose Print Biometrics," in *2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, IEEE, Nov. 2021, pp. 1–6. doi: 10.1109/HNICEM54116.2021.9731994.

[25] S. Dg. Boncolmo, E. V. Calaquian, and M. V. C. Caya, "Gender Identification Using Keras Model Through Detection of Face," in *2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, IEEE, Nov. 2021, pp. 1–6. doi: 10.1109/HNICEM54116.2021.9731814.

[26] M. J. Y. Sutayco and M. V. C. Caya, "Identification of Medicinal Mushrooms using Computer Vision and Convolutional Neural Network," in *2022 6th International Conference on Electrical, Telecommunication and Computer Engineering (ELTICOM)*, IEEE, Nov. 2022, pp. 167–171. doi: 10.1109/ELTICOM57747.2022.10038007.

[27] V. E. Bugayong, J. Flores Villaverde, and N. B. Linsangan, "Google Tesseract: Optical Character Recognition (OCR) on HDD / SSD Labels Using Machine Vision," in *2022 14th International Conference on Computer and Automation Engineering (ICCAE)*, IEEE, Mar. 2022, pp. 56–60. doi: 10.1109/ICCAE55086.2022.9762440.

[28] Ma. K. Agbulos, Y. Sarmiento, and J. Villaverde, "Identification of Leaf Blast and Brown Spot Diseases on Rice Leaf with YOLO Algorithm," in *2021 IEEE 7th International Conference on Control Science and Systems Engineering (ICCSSE)*, IEEE, Jul. 2021, pp. 307–312. doi: 10.1109/ICCSSE52761.2021.9545153.

[29] C. O. Manlises, J. B. Santos, P. A. Adviento, and D. A. Padilla, "Expiry Date Character Recognition on Canned Goods Using Convolutional Neural Network VGG16 Architecture," in *2023 15th International Conference on Computer and Automation Engineering (ICCAE)*, IEEE, Mar. 2023, pp. 394–399. doi: 10.1109/ICCAE56788.2023.10111352.

[30] A. M. N. Antolin, D. A. Padilla, and J. M. D. Reyes, "Tooth Condition Classification for Dental Charting Using Convolutional Neural Network and Image Processing," in *2021 6th International Conference on Image, Vision and Computing (ICIVC)*, IEEE, Jul. 2021, pp. 174–180. doi: 10.1109/ICIVC52351.2021.9527029.

[31] N. C. Ibarra, M. P. Rivera, and C. O. Manlises, "Determination of Leaf Degradation Percentage for Banana leaves with Panama Disease Using Image Segmentation of Color Spaces and OpenCV," in *2023 15th International Conference on Computer and Automation Engineering (ICCAE)*, IEEE, Mar. 2023, pp. 269–275. doi: 10.1109/ICCAE56788.2023.10111455.

[32] Y. Yuan, Y. Li, and H. Bao, "An Empirical Study on the Lane-Change Duration of Naturalistic Driving Based on Multiple Linear Regression Model," in *2020 IEEE International Conference on Artificial Intelligence and Information Systems (ICAIIS)*, IEEE, Mar. 2020, pp. 381–387. doi: 10.1109/ICAIIS49377.2020.9194893.