Melon Ripeness Determination Using K-nearest Neighbor Algorithm

1st Homer John M. Samar
School of Electrical, Electronics, and Computer Engineering
Mapúa University
Manila, Philippines
hjmsamar@mymail.mapua.edu.ph

2nd Hernanny Jeremy J. Manalang
School of Electrical, Electronics, and Computer Engineering
Mapúa University
Manila, Philippines
hjjmanalang@mymail.mapua.edu.ph

3rd Jocelyn F. Villaverde
School of Electrical, Electronics, and Computer Engineering
Mapúa University
Manila, Philippines
jfvillaverde@mymail.mapua.edu.ph

2024 16th International Conference on Computer and Automation Engineering (ICCAE) | 979-8-3503-7005-8/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICCAE59995.2024.10569923

***Abstract*—This paper presents a method for determining the ripeness of Cantaloupe using a K-Nearest Neighbors (KNN) Algorithm on a Raspberry PI. One of the most common problems is determining fruit ripeness purely by visual inspection and traditional methods, such as relying on touch, which is challenging to implement. The Color Segmentation Algorithm used in the study operates in the HSV color space. The Canny Edge detection technique utilizes a region-growing approach, region merging, and initial seed selection. Following the segmentation process, the ripeness of the Cantaloupe is determined using the K-Nearest Neighbors (KNN) Algorithm based on its features, where accuracy reports from the dataset determine the best value of K. The proposed Color Segmentation Algorithm successfully segments the captured Cantaloupe images without any errors and determines their ripeness in most cases based on the KNN Algorithm. However, there are instances where the KNN algorithm incorrectly predicts ripeness from uneven lighting and objects detected in the image, resulting in an accuracy of 80 percent. In general, the system's accuracy based on the Confusion Matrix testing dataset is 95 percent, and as for actual testing, it's 80 percent, as stated before.**

***Index Terms*—color segmentation, edge detection, region growing, region merging, KNN, HSV, Cantaloupe**

I. Introduction

When a person decides to pick or buy a melon, choosing the ones that are not ripe is discouraging since most people favor melons that they can quickly consume and put on display to mature for a week or two [1]-[4]. Fruits in the gourd family, like watermelon and squash, tend to give out clues such as tendrils being directly opposite to the point where the gourd is attached to the vine or the vine becoming hard and woody and even with the texture or smell.[1] [5]. Many of us depend on feeling the melon by touch to see if it is ripe, but it can be discouraging because it often results in the melon not being ripe at all and further damaging the fruit [5]. Since melons like the Cantaloupes do not produce any tendrils like the other fruits in the gourd family, the other clue in indicating when the melon is ripe is by its visible features such as the color, the spots/dots caused by rotting, as well as the net development of the melon's skin [1] [6] [7]. Today's technology, including image processing, makes it possible to use the markers above to determine the ripeness of a melon in a way convenient for the user [8]-[10]. This study aims to use the color segmentation algorithm in image processing using a Raspberry Pi device to determine the ripeness of the cantaloupe fruit. The said device has proven to be capable of the same task as any computer today, which was proven by research done by [11], where a Raspberry Pi device was used to classify red blood cells according to their shapes. The procedure of color segmentation can be described as dividing a picture into useful areas depending on color properties [12]. Fruit with raw skin and ripe skin can have drastically differing skin tones since various color spaces can be used to determine whether a fruit is ripe or just beginning to ripen [13]. The fruit undergoes a variety of changes as it ripens, such as a change in peel color and a change in texture (net development of the skin), which is used to assess the fruit's health and measure its level of maturity without causing damage [14]. Using a technique called color segmentation, a colored image is divided into numerous color-based clusters, and with this, crop maturity can be assessed using a variety of color segmentation approaches, including edge detection, region growth, histograms, and machine learning methods like Support Vector Machine (SVM) and OpenCV and more [15]-[18].

Judging fruit's ripeness only by visual examination is one of the most common problems, and as a study reported in [5][19] demonstrates, conventional techniques of doing so are problematic and difficult to employ, especially in large-scale manufacturing. Additionally, as stated in the first paragraph, the image processing method contains a wide range of techniques, including color segmentation, picture segmentation, and more. However, to make use of these approaches, we must investigate each pixel's feature, which is computationally difficult and takes more time but results in more accurate and reliable results [19] [20].

Other research projects were done regarding the fruit's ripeness, but they were done using common approaches for color segmentation. Even so, there is a gap in the previous research in which a region-growing approach for color segmentation is not involved in fruit ripeness. Furthermore, the said approaches can work together to achieve the final data, such as the use of histogram and then the use of thresholding and more. There are also no further research projects that use Raspberry Pi that involve a region-growing approach and that are tested in the Cantaloupes.

The general objective of the research is to determine melon ripeness using the K-nearest neighbor algorithm. There are four specific objectives to achieve the main objectives: (a) to develop a Raspberry Pi system that uses a Raspberry Pi Camera Module to capture a JPG or PNG image of a cantaloupe., (b) to apply the proposed color segmentation algorithm that involves the region growing approach to the image (Initial Seed Selection using Canny Edge Detection, Region Growing and Region Merging), (c) to determine ripeness of the segmented image through K-Nearest Neighbor (KNN) based on the visible features (skin color, netting, and dots/spots) and lastly is (d) to evaluate the Confusion Matrix's accuracy in identifying the status of ripeness.

The significance of this research lies in its ability to aid individuals who employ traditional methods in differentiating between ripe and unripe melons. In addition to preventing fruit damage, this is advantageous for large-scale production and shoppers looking for melons. Considering this, it can save money for the public as well as for manufacturing businesses, as it would be wasteful to purchase and deliver goods that are not yet ready for sale. This helps reduce difficulty and encourages customers or businesses to choose a melon they prefer because most people choose melons that they can eat right away or even put on display to ripen for a week or two. In addition, the proposed color segmentation algorithm that involves a region-growing approach will be evaluated on cantaloupes and may help other researchers who want to apply the algorithm to other fruits.

The study is mainly focused on detecting the ripeness of the melon Cantaloupe based only on the visible features of the melon, such as the color of the skin, the spots/dots caused by rotting, and the netting that's developed on its skin. Textures of the melon, such as bumps and lines, are not included in this research since they would not affect the melon's ripeness. In addition, the output of the system will only be the classified class of Cantaloupe since the said approaches of color segmentation can work together to achieve the final data, such as the use of histograms, thresholding, and more. The study will involve the approach region growth along with other approaches (Initial Seed Selection using Canny Edge Detection and Region Merging). Lastly, K- nearest neighbor (KNN) is used as a classifier in detecting the ripeness of the fruit melon. In terms of training and testing of data, the dataset will be coming from the internet, such as Kaggle, which consists of 492 images, where 100 images are used for testing and 392 images for training. For the prototype, a Raspberry Pi device will be used as a single-board computer; a Pi camera will be used to capture the image of the melon and an LCD will be used to display the output (ripe/not ripe).

II. Methodology

A. Conceptual Framework

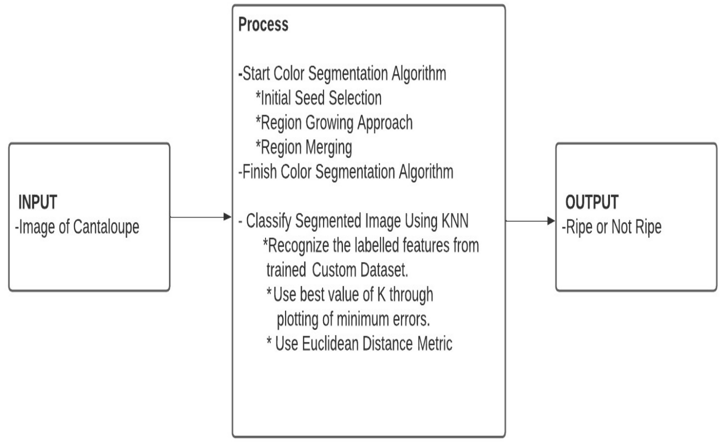

Fig. 1. Conceptual Framework

The conceptual framework of the study calls for inputting a digital image into the system. Using the HSV color space and the suggested Color Segmentation Algorithm, the system finds and processes the image. Canny Edge Detection's edge data is used to carry out automatic seed selection. The next steps involve region growing and region merging. The classifier receives the segmented color image and applies the K-Nearest Neighbor (KNN) algorithm to estimate Cantaloupe's ripeness based on variables including skin color, netting, and dots/spots. The output (ripe/not ripe) that is shown correlates to the melon's ripeness class.

B. Prototype Block Diagram

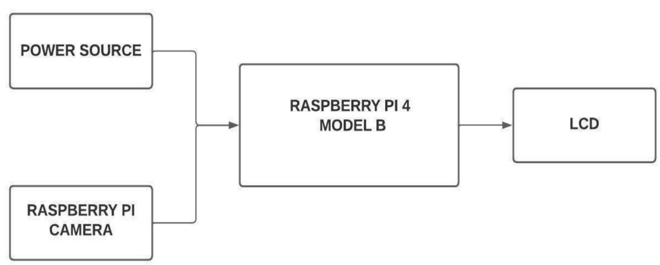

Fig. 2. Prototype Block Diagram

The diagram above displays the system hardware, which consists of an LCD display, a Raspberry Pi 4 Model B, a power source, and a Raspberry Pi camera linked to a breadboard in the prototype block diagram shown above. The Raspberry Pi 4 Model B was selected since it is the newest and most widely utilized microcontroller in computer engineering applications. The Raspberry Pi Camera was chosen because it works well with the Raspberry Pi. The melon's ripeness is shown on the LCD according to its class. Finally, the Raspberry Pi 4 Model B is powered by a power source.

C. System Flowchart

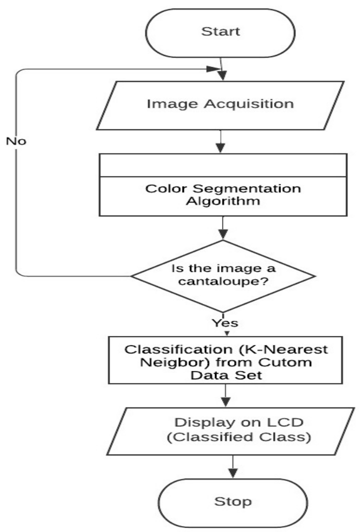

Fig. 3. System Flowchart

The initial picture of the Cantaloupe in the system is taken by the Raspberry Pi Camera Module. The image is changed into HSV color space by the Color Segmentation Algorithm. The region-growing approach's initial seed selection is carried out utilizing Canny Edge detection to establish the starting point. Then, to prevent over-segmentation, regions are combined based on their distance [21]-[23]. Since the algorithm is taught to recognize traits of ripe and unripe cantaloupes, the acquired image must be of a cantaloupe for classification to be accurate. The K- Nearest Neighbor (KNN) algorithm evaluates skin color, netting, and dots/spots to assess cantaloupe maturity. The Cantaloupe's classification is shown on the LCD.

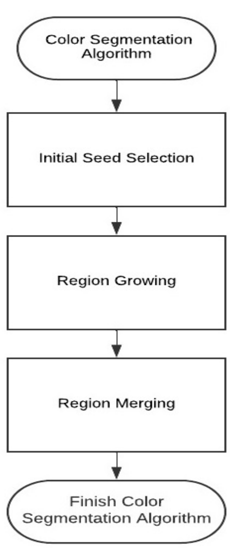

Fig. 4. Color Segmentation Algorithm Flow Chart

The color segmentation algorithm's flow chart illustrates how it operates. Initial Seed Selection uses the Canny Edge detection approach to identify the threshold value that will prevent too many edges[21]-[23]. The image is then transformed into an HSV color space, and the values of the color components (H, S, and V) are computed using a 3x3 neighborhood mask.

The method for calculating the distance E in steps is defined by equations (1) through (4). The edge threshold (TE) is derived using equation (5), which also calculates the average (average) and standard deviation (std) of E. Non-edge pixels are used as the first seeds in TE to acquire the starting point for region growth.

$$\Delta H(i,j) = Hmean(i,j) - H(i,j) \tag{1}$$

$$\Delta S(i,j) = Smean(i,j) - S(i,j) \tag{2}$$

$$\Delta V(i,j) = Vmean(i,j) - V(i,j) \tag{3}$$

$$\Delta E(i,j) = \sqrt{\Delta H(i,j)^2 + \Delta S(i,j)^2 + \Delta V(i,j)^2} \tag{4}$$

$$TE = avgE - 0.7 * stdE; if(avgE - 0.7 * stdE) > 0 \tag{5}$$

$$= avgE; otherwise$$

According to their closeness of recognized pixels, unlabelled pixels on the margins of the image are classified using the region-growing method. The unlabelled pixel is given the label of its distantly closest neighbor. A new label is applied if a similar region cannot be discovered and the distance to all neighbors is greater than the threshold. Until every pixel is labelled, this process is repeated. For the revised regions, the mean values of the color components (Hmean, Smean, and Vmean) are recalculated. Using region merging, over-segmentation is addressed. Equations (6) and (7) are used to compare the distances between the various regions. The neighboring regions are combined into a single region if the threshold is less than the distances.

$$D(l,k) = \sqrt{(al,ak)^2 + (bl,bk)^2} \tag{6}$$

$$D(l,k) = \sqrt{(Hl - Hk)^2 + (Sl - Sk)^2 + (Vl - Vk)^2} \tag{7}$$

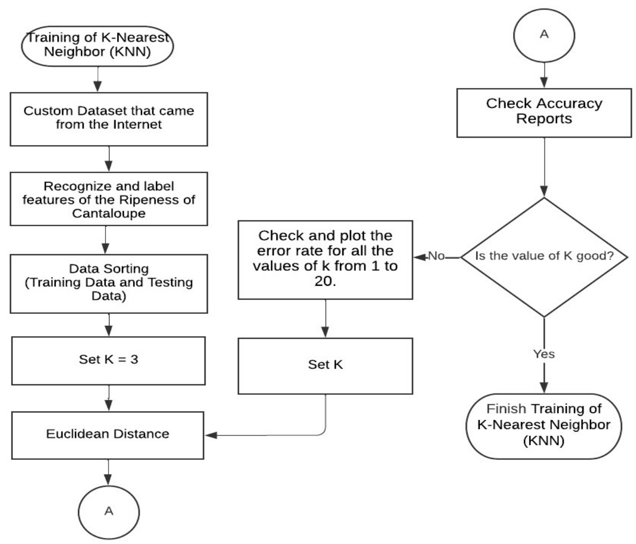

Fig. 5. Training Of K-Nearest Neighbor (KNN)

Given that the study's main objective is to use color segmentation based on the melon's visual features to identify the maturity of Cantaloupe (color of the skin, the spots/dots caused by rotting, and the netting that's developed on its skin), the K- Nearest Neighbor (KNN) is used. Four hundred ninety-two photos from a custom dataset are gathered from sites like Kaggle and manually reviewed for proper labeling. One hundred testing photos and 392 training images make up the dataset. Accuracy reports are assessed with K set to 3 and Euclidean distance as the metric shown in equation (8)[24]. With K values ranging from 1 to 20, training and testing data are used to calculate the ideal K value, which is then used to visualize the error rate. The new K value is the one with the lowest amount of inaccuracy.

$$Dn = \sqrt{\Sigma k |I_l - W_t|^2} \tag{8}$$

D. Test Setup

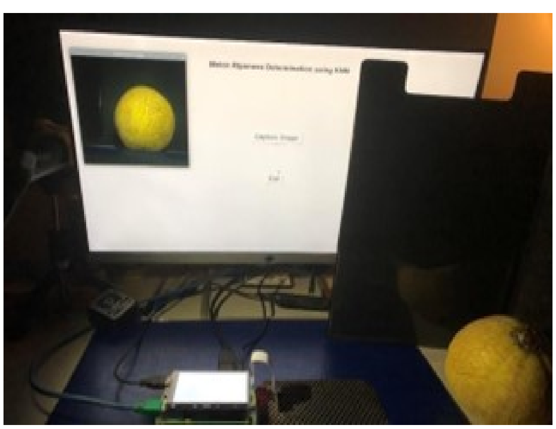

Fig. 6. Actual Experimental Set Up with Artificial Lighting and Black Background

E. Data Gathering

Fig. 7. Sample Cantaloupes to be tested

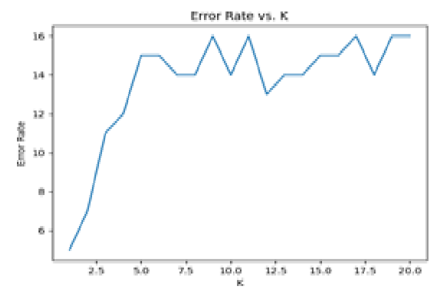

Fig. 8. Finding Best K from Trained Data Set

The provided figures show the 5 Cantaloupes to be used in testing the prototype itself, as well as the plot of the accuracy attained while training the custom dataset with the K-nearest Neighbor (KNN) model to determine the ideal K value. Following training, it is found that K value 1 has the lowest level of inaccuracy, making it the best option for classifying the ripeness of input photos.

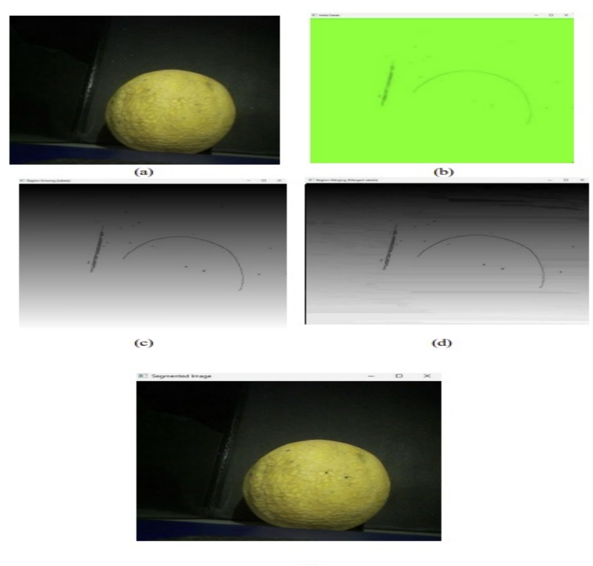

Fig. 9. Each Step of The Proposed Color Segmentation Algorithm (a – Original Image, b- Initial Seed (green area), c-Region Growing, d- Region Merging, and e-Segmented Image)

The color segmentation algorithm divides an image into discrete sections using a multi-step procedure. It begins by utilizing the Canny edge detection method (shown in green) to find the initial seed pixels. After that, it expands regions by assigning complementary hues to nearby pixels. To lessen fragmentation, it then combines sections with comparable color characteristics. Lastly, merged regions are colored using the original image as a guide to create the segmented image required for categorization.

III. Results and Discussion

Based on 100 testing images from the testing dataset, the Confusion Matrix table shows a model accuracy of 95% and displays the distribution of predicted labels in comparison to actual labels.

TABLE I
CONFUSION MATRIX OF TESTING DATASET (KNN)

	Ripe	***Not Ripe***
Ripe	56	3
Not Ripe	2	39

$$Accuracy = \frac{number of correct predictions}{total predictions} * 100 \tag{9}$$

$$Accuracy = \frac{56+39}{56+3+39+2} * 100 = 95\%$$

TABLE II
TESTING TABLE OF ACTUAL SAMPLES

	Actual Class	***Predicted Class***
1	Ripe	Ripe
2	Ripe	Ripe
3	Not Ripe	Not Ripe
4	Not Ripe	Not Ripe
5	Not Ripe	Ripe
6	Ripe	Ripe
7	Ripe	Ripe
8	Ripe	Ripe
9	Not Ripe	Not Ripe
10	Not Ripe	Not Ripe
11	Not Ripe	Not Ripe
12	Ripe	Ripe
13	Ripe	Not Ripe
14	Ripe	Ripe
15	Not Ripe	Ripe
16	Not Ripe	Not Ripe
17	Not Ripe	Not Ripe
18	Ripe	Ripe
19	Ripe	Ripe
20	Ripe	Ripe
21	Not Ripe	Ripe
22	Not Ripe	Not Ripe
23	Ripe	Not Ripe
24	Not Ripe	Not Ripe
25	Ripe	Ripe

TABLE III
CONFUSION MATRIX OF TESTING TABLE

	Ripe	***Not Ripe***
Ripe	11	2
Not Ripe	3	9

$$Accuracy = \frac{number of correct predictions}{total predictions} * 100 \tag{10}$$

$$Accuracy = \frac{20}{25} * 100 = 80\%$$

IV. Conclusion

This paper describes an approach based on K-nearest Neighbor Algorithm that uses the Color Segmentation Algorithm to the input image to determine the Cantaloupe ripeness. The Color Segmentation Algorithm seeks to classify pixels with comparable properties into useful regions. Edge detection is not expressly used in it. Instead, it works by developing regions incrementally from seed sites according to predetermined parameters. The researcher created a system in which it detects the Cantaloupe, applies the proposed Color Segmentation Algorithm, and determines its ripeness based on the color, spots, and netting. The K-nearest Neighbor Algorithm heavily relies on the color since netting and spots and dots are also visible on ripe and not ripe cantaloupes. Through actual testing, the system was able to determine the ripeness of the Cantaloupe. The accuracy of the KNN based on the Confusion Matrix of the testing dataset is 95% based on the results of 100 images. During the actual testing, 20 out of 25 have the correct predicted class. Furthermore, some factors affect the results due to lighting and other objects detected in the image. Generally, the estimated accuracy of the actual testing is 80%.

V. Recommendations

In this study, the researchers only focused on determining the ripeness of Cantaloupe on the color-segmented image based on the color, spots, and netting. Due to lighting and other objects detected in the image, it affects the results of determining the ripeness. The researchers recommend applying a black background and artificial lighting when capturing the Cantaloupe. In addition, the study could also be improved by using a high-quality camera and adding more images of Cantaloupe in the dataset preferably from the Raspberry Pi Camera module since similar characteristics, such as lighting conditions, resolution, and probable artifacts, will be present in images shot with the camera module. Black background and lighting are used since the input image is converted to a binary image, and lighting brings out the netting and spots. Furthermore, KNN is a straightforward and efficient algorithm, but it may not always be the best option for difficult picture classification tasks. Since netting, spots, and dots can be seen on cantaloupes that are ripe and not ripe, the K-nearest Neighbor Algorithm largely relies on color. Deep learning models such as convolutional neural networks (CNNs) have shown exceptional performance in photo categorization tasks since they can automatically acquire structural features. Looking into CNNs or other deep learning models may be useful if researchers are analyzing a lot of data.

References

[1] H. Abidzar Tawakal and A. Prayoga, "The Development of Methods for Detecting Melon Maturity Level Based on Fruit Skin Texture Using the Histogram of Oriented Gradients and the Support Vector Machine," 2019 Fourth International Conference on Informatics and Computing (ICIC), 2019, pp. 1- 4, doi: 10.1109/ICIC47613.2019.8985953.

[2] M. Phothisonothai, S. Tantisatirapong and A. Aurasopon, "Automated determination of watermelon ripeness based on image color segmentation and rind texture analysis," 2016 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), 2016, pp. 1-5, doi: 10.1109/ISPACS.2016.7824766.

[3] J. P. O. Gabriel, M. K. R. Cabunilas and J. F. Villaverde, "Cantaloupe Ripeness Detection Using Electronic Nose," 2022 14th International Conference on Computer and Automation Engineering (ICCAE), 2022, pp. 44-49, doi: 10.1109/ICCAE55086.2022.9762434.

[4] K. Kipli et al., "Image Processing Mobile Application For Banana Ripeness Evaluation," 2018 International Conference on Computational Approach in Smart Systems Design and Applications (ICASSDA), 2018, pp. 1-5, doi: 10.1109/ICASSDA.2018.8477600.

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

[6] N. M. S. Iswari, Wella and Ranny, "Fruitylicious: Mobile application for fruit ripeness determination based on fruit image," 2017 10th International Conference on Human System Interactions (HSI), Ulsan, Korea (South), 2017, pp. 183-187, doi: 10.1109/HSI.2017.8005025

[7] S. O. Nelson, S. Trabelsi and S. J. Kays, "Correlating Dielectric Properties of Melons with Quality," 2006 IEEE Antennas and Propagation Society International Symposium, 2006, pp. 4849-4852, doi: 10.1109/APS.2006.1711729

[8] S. Muhali and N. B. Linsangan, "Classification of Lanzones Tree Leaf Diseases Using Image Processing Technology and a Convolutional Neural Network (CNN)," 2022 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET), 2022, pp. 1-6, doi: 10.1109/IICAIET55139.2022.9936833.

[9] D. M. Dumaliang, J. M. Q. Rigor, R. G. Garcia, J. F. Villaverde and J.R. Cuñado, "Coin Identification and Conversion System using Image Processing," 2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), 2021, pp. 1-6, doi: 10.1109/HNICEM54116.2021.9732002.

[10] M. J. G. Sinco, N. B. Linsangan and R. V. Pellegrino, "DentIS: Dental Bracing Procedure Information System Using Cosine Metric Algorithm for Image Processing," 2019 IEEE 11th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), 2019, pp. 1-5, doi: 10.1109/HNICEM48295.2019.90728

[11] R. V. Pellegrino, A. C. Tarrobago and D. L. B. Zulueta, "Automated RBC Morphology Counting and Grading Using Image Processing and Support Vector Machine," 2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), 2021, pp. 1-5, doi: 10.1109/HNICEM54116.2021.9731937.

[12] Feng Su and Gu Fang, "Chroma-based color enhancement for improved color segmentation," 2015 9th International Conference on Sensing Technology (ICST), 2015, pp. 162-167, doi: 10.1109/ICSensT.2015.7438384.

[13] A. Septiarini, H. Hamdani, H. R. Hatta, and A. A. Kasim, "Image-based processing for ripeness classification of oil palm fruit," 2019 5th International Conference on Science in Information Technology (ICSITech),2019, pp. 23-26, doi: 10.1109/ICSITech46713.2019.8987575

[14] M. N. Kulkarni and R. P. Mudhalwadkar, "Data-driven modeling for banana ripeness assessment," 2017 International Conference on Intelligent Computing and Control Systems (ICICCS), 2017, pp. 662- 665, doi: 10.1109/ICCONS.2017.8250545.

[15] Z. Al-Mashhadani and B. Chandrasekaran, "Autonomous Ripeness Detection Using Image Processing for an Agricultural Robotic System," 2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 2020, pp. 0743- 0748, doi: 10.1109/UEMCON51285.2020.9298168.

[16] 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," 2019 5th International Conference on Control, Automation and Robotics (ICCAR), 2019, pp. 880- 883, doi: 10.1109/ICCAR.2019.8813346

[17] D. A. Padilla, G. V. Magwili, A. L. A. Marohom, C. M. G. Co, J. C. C. Gaño and J. M. U. Tuazon, "Portable Yellow Spot Disease Identifier on Sugarcane Leaf via Image Processing Using Support Vector Machine," 2019 5th International Conference on Control, Automation and Robotics (ICCAR), 2019, pp. 901-905, doi: 10.1109/ICCAR.2019.8813495.

[18] A. M. M. De Los Reyes, A. C. A. Reyes, J. L. Torres, D. A. Padilla and J. Villaverde, "Detection of Aedes Aegypti mosquito by digital image processing techniques and support vector machine," 2016 IEEE Region 10 Conference (TENCON), 2016, pp. 2342-2345, doi: 10.1109/TENCON.2016.7848448

[19] 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," 2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), 2020, pp. 1-6, doi: 10.1109/HNICEM51456.2020.9400023

[20] G. L. Foresti and F. A. Pellegrino, "Automatic visual recognition of deformable objects for grasping and manipulation," in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 34, no. 3, pp. 325-333, Aug. 2004, doi: 10.1109/TSMCC.2003.819701

[21] P. R. Narkhede and A. V. Gokhale, "Color image segmentation using edge detection and seeded region growing approach for CIELab and HSV color spaces," 2015 International Conference on Industrial Instrumentation and Control (ICIC), 2015, pp. 1214-1218, doi: 10.1109/IIC.2015.7150932.

[22] H. Chen, H. Ding, X. He, and H. Zhuang, "Color image segmentation based on seeded region growing with Canny edge detection," 2014 12th International Conference on Signal Processing (ICSP), 2014, pp. 683-686, doi: 10.1109/ICOSP.2014.7015090

[23] M. Mary Synthuja Jain Preetha, L. Padma Suresh and M. John Bosco, "Image segmentation using seeded region growing," 2012 International Conference on Computing, Electronics and Electrical Technologies (ICCEET), 2012, pp. 576-583, doi: 10.1109/ICCEET.2012.6203897.

[24] A. R. Lubis, S. Prayudani, Al-Khowarizmi, Y. Y. Lase, and Y. Fatmi, "Similarity Normalized Euclidean Distance on KNN Method to Classify Image of Skin Cancer," 2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Yogyakarta, Indonesia, 2021, pp. 68-73, doi: 10.1109/ISRITI54043.2021.9702826.