Computer-Vision-Based Detection and Monitoring System for Mature Coconut Fruits with a Web Dashboard Visualization Platform

Samfford S. Cabaluna, Maria Fe P. Bahinting, Leah A. Alindayo
Computer Applications Department-College of Computer Studies,
Mindanao State University Iligan Institute of Technology, Iligan City, Northern Mindanao, Philippines

***Abstract*—The Philippines is the second largest producer of coconut products in the world with 347 million trees planted in 3.6 million hectares of land across the country. Traditionally, harvesting coconuts is a labor-intensive process in the Philippines that involves manual climbing and chopping fruits, which carries a high risk of harm or even death. Hence, the number of expert coconut climbers has decreased as a result. In response, current research has concentrated on creating robot harvesters. However, classifying the mature coconut fruit is a major problem in the harvesting process that calls for a great deal of experience, patience, and work. Studies employing Convolutional Neural Networks (CNNs) have shown great accuracy in detecting coconut ripeness, although these efforts have been limited to detection without practical integration with harvesting equipment. Moreover, the present research lacks a comprehensive solution that allows real-time data display and monitoring, such as the maturation stage of coconuts, via a web-based dashboard. This discrepancy emphasizes the requirement for systems that can not only identify the age of coconuts but also work with harvesting technologies and provide intuitive user interfaces for data display and decision-making. In order to fill these gaps, this study presents a computer-vision-based system that monitors and detects coconut fruit maturity, with an emphasis on mature coconuts, by utilizing the YOLOv8 model. With a Mean Average Precision (mAP50) of 99.5%, mAP50-95 of 89.5%, precision of 99.5%, and recall of 99.9%, the system demonstrated great accuracy. A web-based dashboard is also integrated into the system to provide monitoring and visualization of detected coconut fruits, along with notifications for fully ripe fruits.**

***Keywords*—Coconut fruit maturity; coconut maturity detection; computer vision; crop monitoring**

I. Introduction

The Philippines is the world's second-largest exporter of coconuts, with 14.89 million metric tons produced in 2023.

With the pressing issues on risks of manual harvesting which caused a steady decline in coconut harvester population, robot coconut harvesters were developed to address the issue based on the review by Kumar et al. [1]. Furthermore, Cousalya et al. [2] created a mobile operated coconut harvesting machine. These developed harvesters, however, lack a detecting systems that resulted to harvest even the young coconuts. Nevertheless, the studies of Sakthipreasad and Megalingam, Junaedy et al., Titus et al., Wibowo et al., and Divyanth et al. [3], [4], [5], [6], [7], on robotic harvesters that are integrated with vision-based detection but it solely focused on fruit recognition which lead to the harvester inability to specifically locate the matured coconuts.

Determining the ripeness of coconut fruit accurately remains a challenging task which has an impact on product quality and customer satisfaction [8]. Conventional manual techniques take a lot of time and require experience [9], [10], [11], [12], [13], and elements like dim lighting and complicated backgrounds [12], [14] make it hard for computers and humans to recognize coconuts [6], [15], which frequently results in the harvesting of poor or premature coconuts. In response, recent research have been conducted that introduces the concept of coconut maturity detection using several methods. Two studies use the fuzzy logic approach in detecting and classifying coconut fruit maturity [9], [12] which proved to be an effective algorithm where in the study of Megalingam et al. uses the Decision-Making Probability (DMP) model for real-time classification achieving an accuracy of 86.3%, but Mask R-CNN performs better than the other integrated models. Aside from these studies, other researches focused on Convolutional Neural Networks (CNNs) in coconut fruit maturity detection and classification which is commonly implemented having an average accuracy above 90%.

Moreover, Venkatesh et al. [14] implemented the SOLO model, Artificial Neural Network is implemented by Hendrawan et al. [16], Anushya implemented K-Means clustering wherein image features are extracted by Gray-level Co-Occurrence Matrix (GLCM) determining the freshness of the coconut obtaining 97% accuracy [17], Varur et al. [18] researched on coconut development phases classifying into five classes using Xception, ResNet50V2, ResNet152V2, and MobileNetV2, wherein MobileNetv2 produced the best accuracy, Subramanian and Sankar used RestNet-50 and Faster R-CNN wherein ResNet-50 achieved a top-1 accuracy for both premature and mature phases achieving a detection score of 99%, and around 98% top-1 and top-5 accuracies [10], [15], Nguyen et al. also discovered that ResNet101 was the most accurate, using binarized saturation pictures achieving an accuracy of 95% [11], Avudai Nayagam and Devakumar used small VGG net and MobileNet which ensured precise coconut detection when used in real-time on NUC/Jetson Nano boards and climbing robots, however their system's gap focuses on how well those models run on those devices stated [19], and Novelero and Dela Cruz's study (Philippines) used the YOLOv5 model in an on-tree mature coconut fruit detection using UAVs achieving an

This paper is sponsored by the Philippine Department of Science and Technology Engineering Research and Development for Technology.

accuracy of 88.4% which will be of great value in eliminating risks of harvesting coconuts in the future since it can also be useful for coconut crop yield estimation, [20]. In the research of Mandava et al. [13], it is concluded that YOLOv5s is superior in metrics yet needing more datasets.

Based on the literature discussed above, the different methods in detecting matured coconut has been successful particularly the utilization of Convolutional Neural Networks (CNN) which incurred a high accuracy rating of above 90% in terms of detecting mature coconut. However, these studies solely focused on maturity detection and was not able to provide a comprehensive solution incorporating real-time display and monitoring of coconut fruits. This emphasizes the necessity for systems that can both identify coconut fruit maturity and integrate it with harvesting technology [20], offering a user-friendly interface for data visualization and informed decision-making.

This paper implements both computer-vision-based detection of coconut fruit maturity using a later model under CNN, the later model, YOLOv8, as it builds upon the success of previous versions and introduces new features and improvements to boost performance and flexibility, applicable as well in different environments with a detailed comparison of its performance metrics to other versions and models [21]. Along with it is an integration with a web dashboard for display and monitoring of scanned coconut fruit maturity data and with a notification system wherein for every detected mature coconut fruit, a notification is sent.

II. Methodology

To design and develop a detection and monitoring system to detect and monitor coconut fruits according to maturity YOLOv8 model was utilized. The researchers as well designed and developed a web dashboard for monitoring the scanned coconut fruits according to maturity. Fig. 1 presents the flow of methods the researchers underwent in coming up with the output of this paper. Having identified the problems as well as motivation to resolve these problems, the researchers then proceeded to the specific objectives consisting of three phases: (Phase 1) the overall design of the system, involving requirements analysis, the design, and the gathering of datasets; (Phase 2) the development of the detection and monitoring systems as well as their integration, involving the processing of the image datasets and their training, and the design and development of the web dashboard for the display of coconut fruit data scanned as well as the integration with the notification system for every mature coconut fruit detected; and (Phase 3) the implementation of the system, involving the testing and evaluation of both detection and monitoring systems.

A. System Requirements Analysis

The following items below serves as guide for the researchers in coming up with the output of this paper. The overall system is limited for experimental purposes:

- Image datasets involve generally both Mature and Premature coconut fruits, wherein Mature coconut fruits include those with both brown and brown-and-green surfaces and Premature coconut fruits include those of young coconut fruits with green surfaces.
- Acquisition of datasets are only limited to close shots angle.
- Web dashboard involves the display of footage of scanned coconut fruits and the display of count data through a line graph chart.
- Count data of coconut fruit maturity are also displayed on the table and can be printed to CSV file.
- The system will send notification mail every time a mature coconut fruit(s) is/are detected.
- The system, limited to experimental use, limits it’s testing within the laboratory, not yet real-time.

Fig. 2 shows the architecture of the overall system, involving both detection and monitoring systems based on the listed system requirements analysis above.

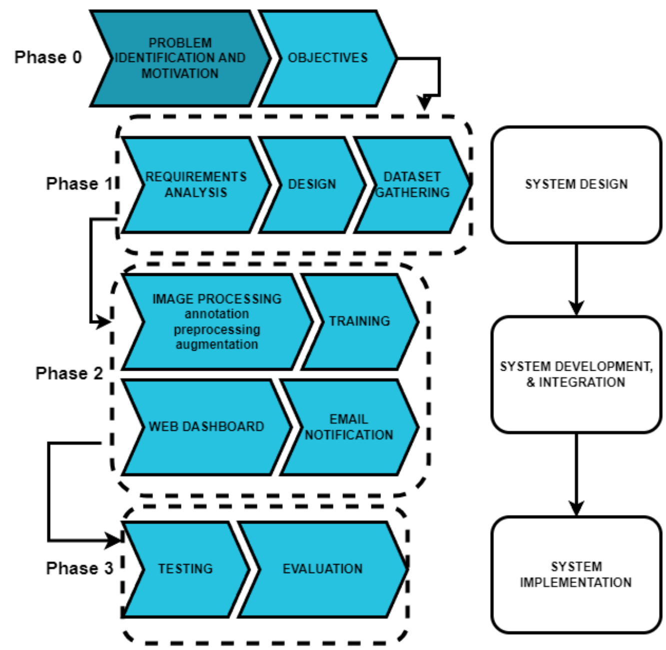

Fig. 1. Methodology flowchart.

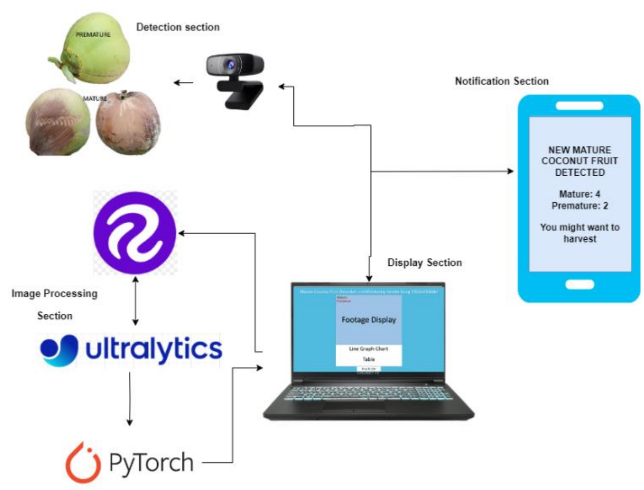

Fig. 2. System architecture.

B. Detection System

This particular section discusses the methods undergone by the researchers in designing and developing the detection system of this paper.

1) Data acquisition: Data were acquired by capturing video footage of harvested coconuts from a farm in Barangay Argayoso, Manticao, Misamis Oriental. Images were extracted at a rate of three per second, resulting in 260 image datasets. These were then uploaded to Roboflow for image processing, including annotation, preprocessing, augmentation, and dataset generation, with two classes: Mature and Premature coconuts (see Fig. 2). Mature coconut fruits include both full brown-colored surface and those with both brown and green surfaces, while the premature coconut fruits include those with green surfaces.

2) Annotation: Following the uploading of images to Roboflow, the platform for annotation was then set up using Smart Polygon (though under the Bounding Box platform) for more precise results. A total of 260 images were annotated, resulting in 409 annotations for the mature class and 403 for the premature class, ensuring a balanced dataset (see Fig. 3).

Fig. 3. Annotation of image datasets.

3) Training of the model: Ultralytics Hub platform was used for the training of the model (see Fig. 4). Google Colab platform was specifically used for the training while the results are automatically uploaded on the Ultralytics dashboard. YOLOv8 model was used for training.

4) Running the trained model: Following the model training is its implementation (see Fig. 5). The researchers have developed a program wherein the scanned coconut fruits according to maturity are counted. Footage is then displayed on the dashboard. Programming of the detection system is done using OpenCV Python.

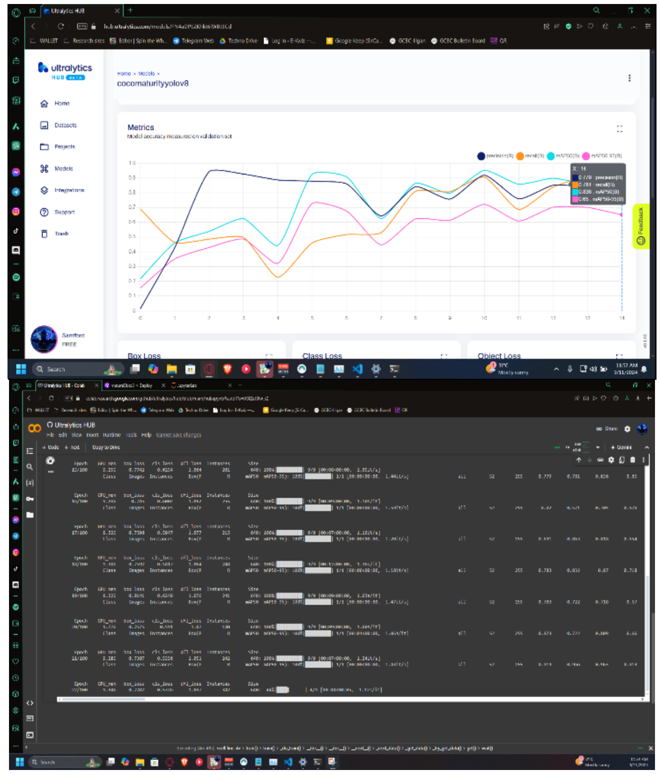

Fig. 4. Training of model.

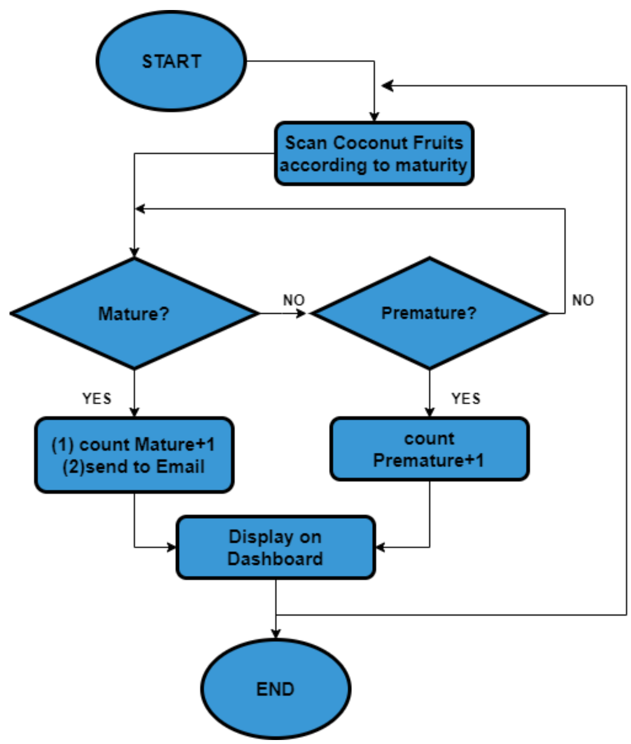

Fig. 5. Detection system flow.

C. Web Dashboard Design

Fig. 6 shows the operational flow of the web dashboard that serves as the display for the overall system. The user will have to go through the homepage where a button will be clicked to enter the dashboard. Once clicked, the user will be led to the main dashboard where the data are displayed.

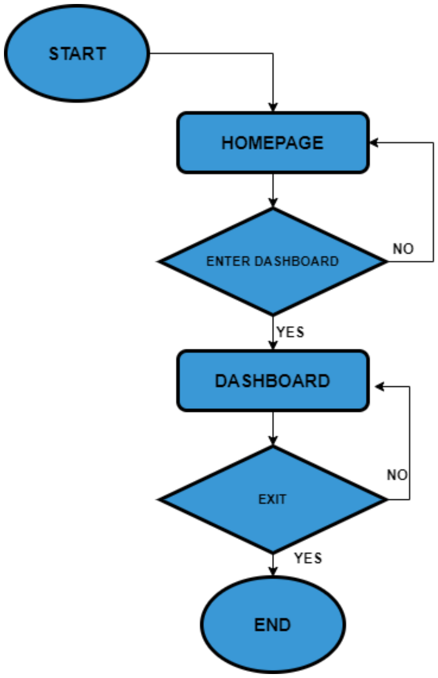

Fig. 6. Web dashboard flow.

D. Integration of Detection System to Web Dashboard and Email

Basic HTML was used in programming the monitoring dashboard with styles for its background image, as well as for the table and charts. The dashboard is then connected to and from the python program in order for the data to be uploaded and displayed on the output of this program. The system is also integrated into the Email for notification purposes.

E. Testing and Evaluation

A sample test run was done for both detection and monitoring systems. As stated earlier, this paper is limited to experimentation, thus the test run was done inside the laboratory.

III. Results

This section presents and discusses the results of this particular paper. Discussion involves the results of training of model as well as the display of data for monitoring.

A. Training Results

After 100 epochs of training, metrics show that the training of the model produced a Mean Average Precision 50 (mAP50) is 99.5% and its mAP50-95 is 89.5%, a 99.5% precision, and 99.9% recall (see Fig. 7) and is gradually increasing.

Fig. 8 presents the results of the training of the model particularly showing the box, class, and object losses.

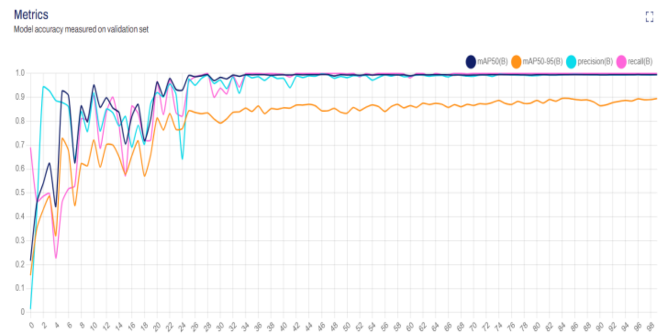

Fig. 7. Training results metrics (mAP, Precision, Recall).

Box Loss shows that at the end of the 100 epochs training, training reaches 46.5% and 53.2% validation. For class loss, training reaches 23.3% while validation reaches 26.8%. Finally, for object loss, training reaches 90.7% and validation reaches at 90.8%.

For the Box Loss, initially, both training and validation loses fluctuate significantly, indicating instability at the start. However, by the end of the graph, both losses converge and stabilize, though the validation loss remains slightly higher than the training loss, indicating that the model generalizes reasonably well but might still benefit from further refinement. This pattern is typical in model training, where validation loss may fluctuate more than training loss due to unseen data.

For the Class Loss, both training and validation losses are initially high, especially around iteration 0, indicating a poor classification performance at the start of training. The training decreases quickly, showing that the model improves its class predictions as training progresses. The validation loss fluctuates significantly, indicating that the model struggles with unseen data early in training. Over time, the validation loss stabilizes, though it remains more variable than the training loss.

Finally, for the Object Loss, both training and validation losses start around 1.2, indicating a relatively high error rate in detecting objects at the beginning of the training. The training loss decreases gradually over time, indicating steady improvement in object detection during training. The curve follows a gentle downward trend, becoming more stable but still showing some fluctuations. The validation loss is initially lower than the training loss and remains relatively stable throughout the training process, hovering around 1.0.

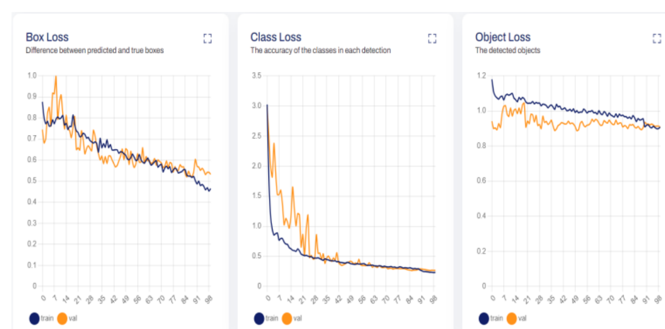

Fig. 8. Training results (Box, Class, Object Losses).

Thus, the model is progressively learning to detect objects better, with both training and validation object losses decreasing and converging. The relatively stable validation loss suggests that the model generalizes well to unseen data in terms of object detection, though there remains room for further improvement as the final loss is still above 0.8. Fig. 9 below presents the confusion matrix based on the results of the training of the model. It particularly presents that 99.5% of actual mature/premature coconuts were correctly detected. 0.5% of predicted mature/premature coconuts were incorrect (wrong predictions). 0.1% of actual mature/premature coconuts were missed (or undetected). Display of detection is presented in Fig. 11.

	PREDICTED POSITIVE (MATURE/PREMATURE)	PREDICTED NEGATIVE (MISSED)
ACTUAL POSITIVE (Mature/Premature)	99.5%	0.1%
ACTUAL NEGATIVE (Not Detected)	0.5%	N/A

Fig. 9. Confusion matrix.

B. Web Dashboard

This particular section discusses the operation of the web dashboard where the data of scanned coconut fruits based on maturity is displayed. Fig. 10 below presents the homepage of the dashboard, wherein the user has to click the ENTER DASHBOARD button in order to enter the main dashboard.

Fig. 10. Dashboard homepage.

The dashboard consists of the footage of the video of the scanning of coconut fruits. The number of coconut fruits scanned based on maturity is displayed as well on the dashboard. The count data are also visualized on the line graph chart and the table as well. The count data of coconut fruits on the dashboard can also be printed to CSV file.

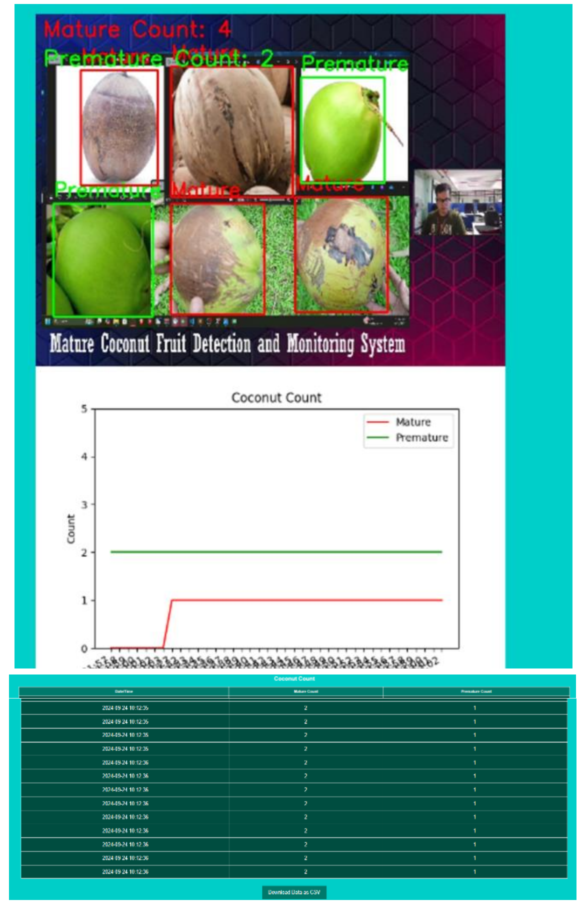

Fig. 11. Monitoring dashboard.

C. Testing and Evaluation of the System

As stated earlier, the overall system was tested and evaluated inside the laboratory since this paper is only limited to experimental level. The overall system was run using an NVIDIA GEFORCE RTX 3060 laptop having a 24GB RAM and 1.5TB internal storage with a 6GB GPU memory. Since the datasets were collected from coconut harvested fruits, therefore, the system can be able to correctly detect the maturity of the coconut fruits when the system's camera is placed very close to the object. Furthermore, with a small amount of datasets wherein mature coconut fruits consists of both brown and brown-and-green surfaces, the system expectedly misclassifies the maturity of coconut fruit. Therefore, more datasets should be added to the training of the model for a more accurate classification of coconut fruits according to maturity.

Fig. 12 below presents the notifications sent to the email for every mature coconut fruit detected. The system will notify through the email every time a new mature coconut fruit is detected.

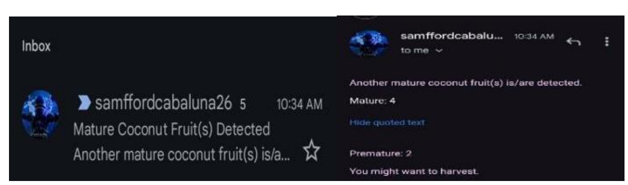

Fig. 12. Email notification.

IV. Discussion

This section discusses in summary the results shown on the previous section, discussing the indications of the figures.

A. Discussion on Training Results

Training results show a gradual increasing in terms of metrics and decreasing in terms of losses, indicating an improving of training results. The overall decrease in both lines for box loss indicates a positive sign of improvement in bounding boxes. The convergence and the stabilizing of losses at a relatively low value for class loss suggests that the model performs well, although it still classifies slightly better on training data than on validation data. Overall, the model's class detection accuracy improves significantly over time, with a more stable training loss, and validation loss eventually converging but with higher variance. There are minor fluctuations for both losses under object loss, but overall, the validation loss shows a stable trend compared to the training loss. Both losses converge toward a similar value below 1.0 by the end of the iterations, suggesting that the model is improving in detecting objects and shows consistent performance on both training and validation datasets. The results shown on the confusion matrix indicate how well-performing the model is in classifying coconut fruit maturity, although there are minor misclassifications, yet the results proved that the model is performing well.

While the model is performing well in classifying coconut fruit maturity, it needs additional image datasets for an even more accurate classification since this paper is limited to close angle shots of the image datasets. Further improvement of such classification requires a consideration of drone shot angles and distances.

B. Discussion on Monitoring System

The monitoring dashboard design prototype displays an outstanding visualization of monitoring the status for coconut fruits in terms of their maturity stages. The notification system through email integrated with it shows that this system can aid in the real-time monitoring of coconut fruits. Since the system is in its experimental stage, this study recommends that for future improvement, the monitoring system will be integrated with an operational mobile app with SMS notification for a more user-friendly monitoring of the app.

V. Conclusions and Recommendation

In conclusion, the researchers have designed and developed, as well as implemented a detection and monitoring system for mature coconut fruits, separating them from premature coconut fruits. The YOLOv8 model proved to be a good model for detection, though more datasets are needed to be added to the training of the model for an even more accurate detection. A web dashboard was also designed and developed, as well as integrated with the detection system, so that the footage of the scanning of coconut fruits are displayed on the dashboard along with the count data, wherein the count data can also be printed to CSV file as well.

Since the model needs more datasets due to some misclassifications occurring in the detection, by way of recommendation, it is highly recommended that there is a certain distance from the camera as well as its lighting resolution to be considered in gathering datasets to prepare the system for deployment with a more accurate detection; on-tree datasets are also highly recommended for training. It is highly recommended as well that the dashboard for display of data will be deployed in a fully operational app. Finally, it is very highly recommended that this system is integrated on a robotic harvester or any automated system by way of modernizing and improving the method of harvesting coconut fruits to aid the harvest of coconut fruits. In addition to increasing coconut harvesting efficiency, this integrated system can help achieve the Sustainable Development Goals (SDGs) of the UN, including Goals 2 (Zero Hunger), 9 (Industry, Innovation, and Infrastructure), and 17 (Partnerships for the Goals).

Acknowledgment

The researchers acknowledge the financial support of the Department of Science and Technology Engineering Research and Development for Technology (DOST ERDT) of the Philippines for this study along with the supervision of the Department of Computer Applications of the College of the Computer Studies of Mindanao State University Iligan Institute of Technology.

References

[1] M. Kumar, G. Bej, and H. S. Pandey, "Status and Scope of Automated Coconut Harvester in India: A Review," Journal of Experimental Agriculture International, vol. 45, no. 5, Art. no. 5, Mar. 2023, doi: 10.9734/jeai/2023/v45i52115.

[2] C. Cousalya, C. Deepalakshmi, V. Keerthika, M. Malini, and P. Arthiya, "Mobile Operated Coconut Harvesting Machine," vol. 4, no. 5, 2021, doi: 10.15680/IJMRSET.2021.0405009.

[3] K. M. Sakthiprasad and R. K. Megalingam, "A Survey on Machine Learning in Agriculture - background work for an unmanned coconut tree harvester," in 2019 Third International Conference on Inventive Systems and Control (ICISC), Jan. 2019, pp. 433–437. doi: 10.1109/ICISC44355.2019.9036375.

[4] A. Junaedy, I. A. Sulistijono, and N. Hanafi, "Particle swarm optimization for coconut detection in a coconut tree plucking robot," 2017 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC), pp. 182–187, Sep. 2017, doi: 10.1109/KCIC.2017.8228584.

[5] A. B. Titus, T. Narayanan, and G. P. Das, "Vision system for coconut farm cable robot," 2017 IEEE International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), pp. 443–450, Aug. 2017, doi: 10.1109/ICSTM.2017.8089201.

[6] T. S. Wibowo, I. A. Sulistijono, and A. Risnumawan, "End-to-end coconut harvesting robot," in 2016 International Electronics Symposium (IES), Sep. 2016, pp. 444–449. doi: 10.1109/ELECSYM.2016.7861047.

[7] L. G. Divyanth, P. Soni, C. M. Pareek, R. Machavaram, M. Nadimi, and J. Paliwal, "Detection of Coconut Clusters Based on Occlusion Condition Using Attention-Guided Faster R-CNN for Robotic Harvesting," Foods, vol. 11, no. 23, Art. no. 23, Jan. 2022, doi: 10.3390/foods11233903.

[8] L. Noppon and P. Nipon, "The Application of Convolution Neural Network for Coconut Maturity Classification," in 2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), May 2021, pp. 112–115. doi: 10.1109/ECTI-CON51831.2021.9454744.

[9] I. M. Javel, A. A. Bandala, R. C. Salvador, R. A. R. Bedruz, E. P. Dadios, and R. R. P. Vicerra, "Coconut Fruit Maturity Classification using Fuzzy Logic," in 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology,Communication and Control, Environment and Management (HNICEM), Nov. 2018, pp. 1–6. doi: 10.1109/HNICEM.2018.8666231.

[10] P. Subramanian and T. S. Sankar, “Detection of maturity stages of coconuts in complex background using Faster R-CNN model,” Biosystems Engineering, vol. 202, pp. 119–132, Feb. 2020, doi: 10.1016/j.biosystemseng.2020.12.002.

[11] N.-L. Nguyen, N. Van Khanh Duong, and H. D. T. Nguyen, “Deep Learning Based Coconut Fruit Maturity Classification,” in Advances in Information and Communication Technology, P. T. Nghia, V. D. Thai, N. T. Thuy, L. H. Son, and V.-N. Huynh, Eds., in Lecture Notes in Networks and Systems. Cham: Springer Nature Switzerland, 2023, pp. 51–59. doi: 10.1007/978-3-031-49529-8_6.

[12] R. K. Megalingam, S. K. Manoharan, and R. B. Maruthababu, “Integrated fuzzy and deep learning model for identification of coconut maturity without human intervention,” Neural Comput & Applic, Jan. 2024, doi: 10.1007/s00521-023-09402-2.

[13] R. K. Mandava, H. Mittal, and N. Hemalatha, “Identifying the maturity level of coconuts using deep learning algorithms,” Materials Today: Proceedings, Sep. 2023, doi: 10.1016/j.matpr.2023.09.071.

[14] Venkatesh, P. K. P, P. Patil, P. G, P. Poojary, and S. B. Basapur, “CRipS: Coconut Ripeness Stage Detection System,” in 2023 International Conference on Network, Multimedia and Information Technology (NMITCON), Sep. 2023, pp. 1–9. doi: 10.1109/NMITCON58196.2023.10276136.

[15] P. Subramanian and T. S. Sankar, “Coconut Maturity Recognition Using Convolutional Neural Network,” in Computer Vision and Machine Learning in Agriculture, Volume 2, M. S. Uddin and J. C. Bansal, Eds., in Algorithms for Intelligent Systems. , Singapore: Springer Singapore, 2022, pp. 107–120. doi: 10.1007/978-981-16-9991-7_7.

[16] Y. Hendrawan, A. Amini, D. Maharani, and S. M. Sutan, “Intelligent Non-Invasive Sensing Method in Identifying Coconut (Coco nucifera var. Ebunea) Ripeness Using Computer Vision and Artificial Neural Network,” 2019. Accessed: Jan. 25, 2024. [Online]. Available: https://www.semanticscholar.org/paper/Intelligent-Non-Invasive-Sensing-Method-in-Coconut-Hendrawan-Amini/cf3d736e20a00ddc238c0940a5161b5db5d92712

[17] Dr. A. Anushya, “Freshness Detection of Coconut via K-Means,” IJRASET, vol. 8, no. 2, pp. 435–438, Feb. 2020, doi: 10.22214/ijraset.2020.2066.

[18] S. Varur, S. Mainale, S. Korishetty, A. Shanbhag, U. Kulkarni, and M. S. M, “Classification of Maturity Stages of Coconuts using Deep Learning on Embedded Platforms,” in 2023 3rd International Conference on Smart Data Intelligence (ICSMDI), Mar. 2023, pp. 343–349. doi: 10.1109/ICSMDI57622.2023.00067.

[19] T. Avudai Nayagam and T. Devakumar, “Intelligent System for Matured Coconut Identification,” IJRTE, vol. 9, no. 1, pp. 57–61, May 2020, doi: 10.35940/ijrte.F9520.059120.

[20] J. M. Novelero and J. C. D. Cruz, “On-tree mature coconut fruit detection based on deep learning using UAV images,” presented at the 2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom), IEEE, 2022, pp. 494–499.

[21] “(PDF) A Review on YOLOv8 and Its Advancements,” in ResearchGate. doi: 10.1007/978-981-99-7962-2_39.