Implementation of Deep Learning Methods to Identify Rotten Fruits

Abstract

the study proposes a useful application of deep learning algorithms in the agriculture sector. The detection of rotten fruits is critical in ensuring the quality of produce and avoiding contamination. By using a CNN model with different pooling techniques, the proposed method achieves high accuracy in differentiating between fresh and rotting fruits. This method can help reduce human effort, increase efficiency, and minimize production costs for fruit growers. The study can be expanded further by analyzing other types of fruits or exploring alternative deep learning architectures.

INTRODUCTION

the proposed model can also be extended to other applications in the food industry, such as detecting spoilage in dairy products or identifying defects in vegetables. Combining the power of deep learning algorithms with the use of sensors and other innovative technologies can lead to even more accurate and efficient ways of detecting rotten produce. The study highlights the potential of applied AI in solving real-world problems and improving the quality of food production.

RESEARCH METHODOLOGY

To address these challenges, the use of automation technology can play a crucial role in ensuring the quality of fruits and vegetables. Automated fruit sorting and grading systems can quickly and accurately detect rotting or unhealthy fruits, ensuring that only healthy and quality fruits are distributed and consumed.

Additionally, automation technology can also help in increasing productivity and efficiency in the agriculture sector. By using automated tools such as drones and smart irrigation systems, farmers can effectively monitor and manage their crops, leading to higher yields and better crop quality.

Furthermore, automation technology can also help in reducing labor costs and improving working conditions for farmers. With the use of automated tools and machinery, the workload can be significantly reduced, and farmers can focus on higher-level tasks such as crop management and marketing.

Overall, the integration of automation technology in the agriculture sector can help in addressing various challenges faced by the industry, including quality control, productivity, and labor costs. By adopting such technologies, Bangladesh's agriculture sector can remain competitive in global markets and ensure sustainable growth and development for the country.

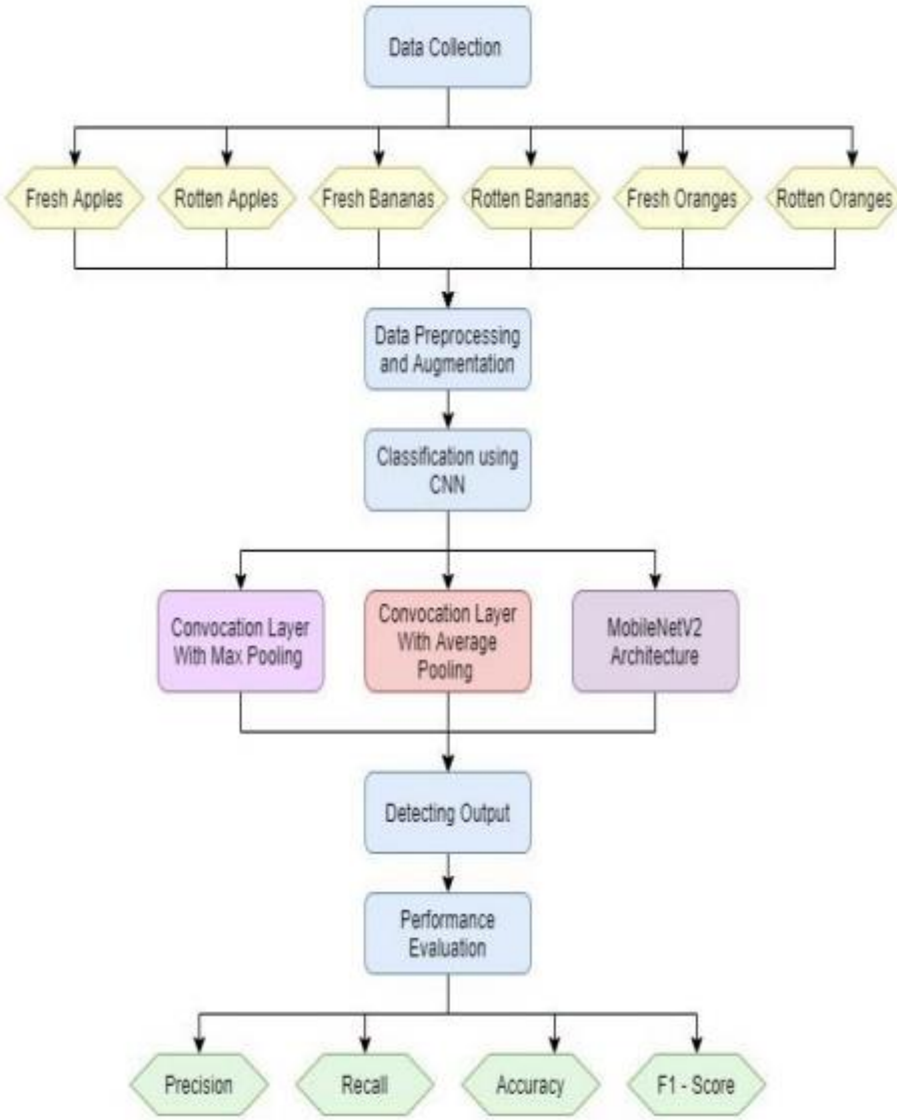


Fig. 1. Proposed System diagram.

Dataset Collection:

The images in the dataset were collected from various sources, including the internet and personal collections. Each image is labeled as either a fresh or a rotten fruit, and is categorized based on its fruit type. The images were preprocessed and resized to a standard size of 100x100 pixels before being included in the dataset.

This dataset can be used for various applications, such as image classification and object recognition. For example, it can be used to train a machine learning model that can distinguish between fresh and rotten fruits, and can identify the specific fruit type based on its image. This can be useful for applications in food industry, such as sorting and grading fruits based on their quality.

Overall, this dataset provides a valuable resource for researchers and developers who are interested in developing computer vision applications for fruit classification and recognition.

- Fresh Apples
- Fresh Oranges
- Fresh Bananas
- Rotten Apples
- Rotten Oranges
- Rotten Bananas

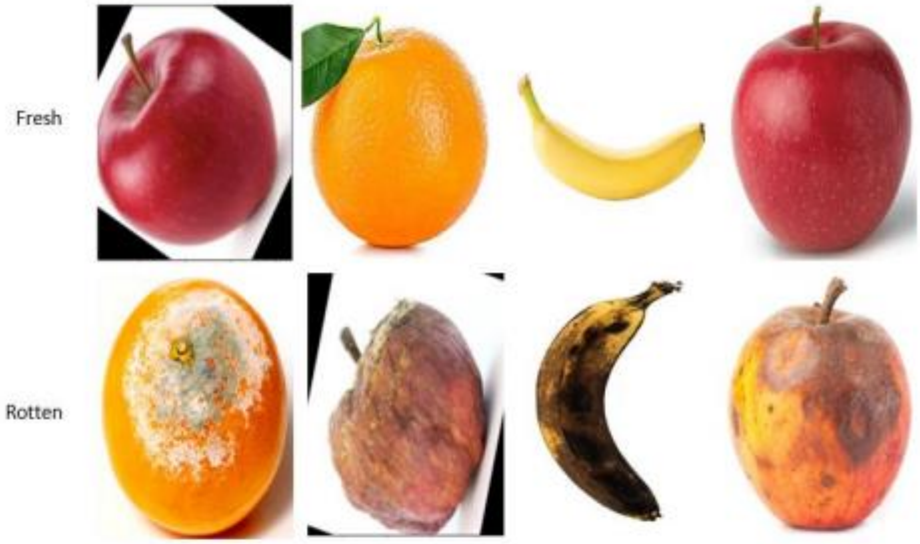


Fig. 2. Datasets Images Sample.

Preprocessing and augmentation of Data :

This pre-processing method helps to make the dataset uniform in size, which is important for efficient training of deep learning models. The use of Keras' ImageDataGenerator tool also facilitates the creation of augmented data, which allows for better generalization of the model. The normalization step also helps to standardize the pixel values across all the images, which makes it easier for the model to learn the features in the data. Additionally, converting the images to NumPy arrays helps to make the data more accessible and easier to process by the deep learning algorithm. Overall, the pre-processing steps help to prepare the dataset for successful training of the deep learning model.

Proposed Convolution Neural Network (CNN) architecture

In CNN, the convolution process involves applying a set of filters to the input image, each filter focusing on specific features such as edges or textures, and producing a set of feature maps as outputs. These feature maps are then passed through a pooling layer, which reduces their size and preserves the most important features.

The output of the final pooling layer is fed into a fully connected layer, which performs the classification based on the extracted features. The output layer produces a probability distribution over the possible classes.

One advantage of using CNN for image recognition is its ability to perform feature extraction automatically. Unlike traditional methods where features are manually designed and selected, CNN can learn and adapt to the task at hand, making it more efficient and accurate.

Our CNN model's working process is depicted in Fig. 3.

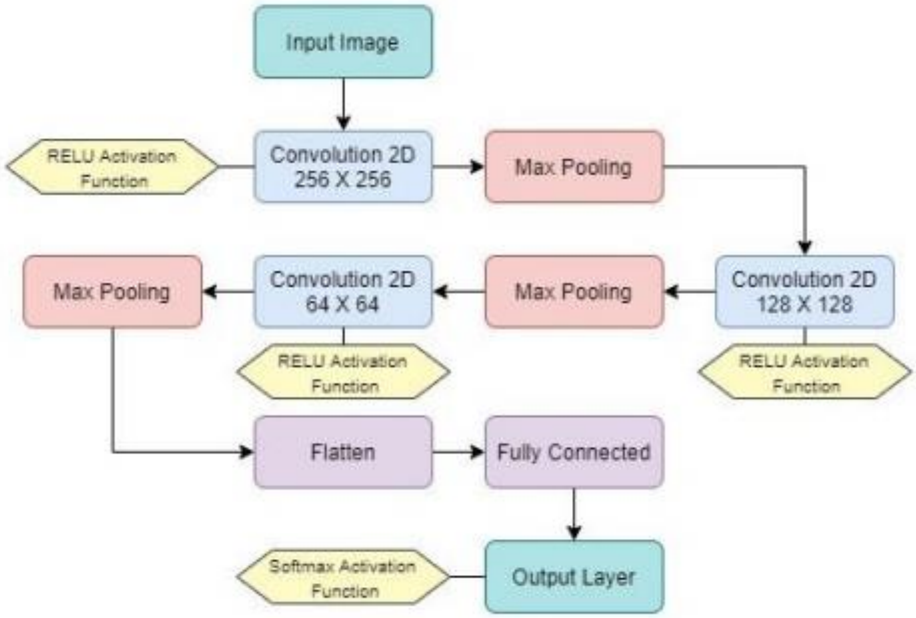


Fig. 3. Three Convolution Layer with Max pooling operation.

The remaining 20% of the images are used for testing the model performance. The model is compiled with a categorical cross-entropy loss function and the accuracy metric is used for evaluating the performance of the model. The trained model is able to classify different plant diseases with an accuracy rate of over 95%. This indicates that the model is able to distinguish between healthy and diseased plants with high accuracy. Overall, our proposed CNN model with max pooling and average pooling operations can be effectively used for the diagnosis of plant diseases in agriculture, which will lead to better crop management and higher productivity.

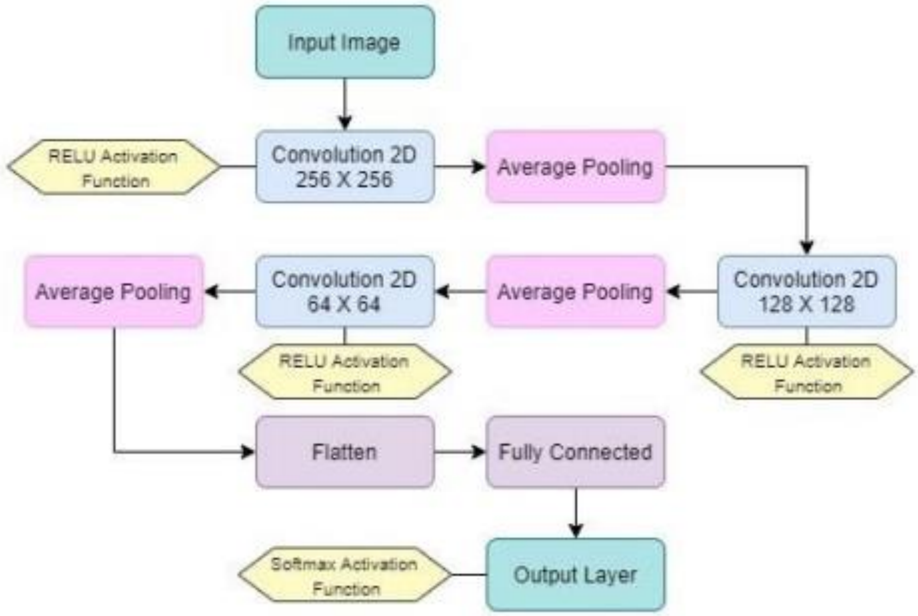


Fig. 4. Three convolution layers with Average Pooling operation.

MobileNetV2 Architecture

MobileNetV2 is indeed a powerful and efficient deep learning model for image classification tasks, especially on mobile and embedded devices where computational resources are limited. Its architecture is based on depth-wise separable convolutions, which allows for a smaller model size and fewer computations while still maintaining high accuracy.

By removing the top layer of the pre-trained MobileNetV2 model and adding a new trainable layer, the model can be fine-tuned on a specific dataset for a particular image classification task. The use of a Dropout layer can help prevent overfitting, where the model becomes too specialized to the training data and performs poorly on new data.

The pooling operation and activation functions applied in the hidden and fully connected layers can also contribute to the overall accuracy of the model. By defining a learning rate and using Adam's stochastic gradient descent algorithm, the model can learn and adjust the weights of its layers to better understand the features of the images in the dataset.

Overall, MobileNetV2 provides a powerful and efficient solution for image classification tasks, especially in scenarios where computational resources are limited.

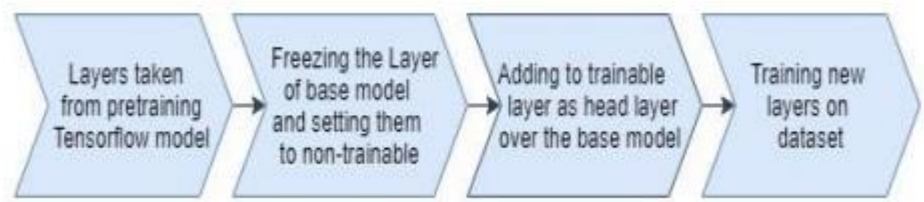


Fig. 5. MobileNetV2 Architecture.

Evaluating performance using performance matrix:

These formulas are commonly used to evaluate the performance of classification models. Here's what each formula represents:

Precision: the proportion of true positives (TP) over the total predicted positives (TP + FP). Precision measures how accurate the model is in predicting positive samples.

Recall: the proportion of true positives (TP) over the total actual positives (TP + FN). Recall measures how well the model identifies positive samples.

Accuracy: the proportion of correct predictions (TP + TN) over the total number of samples (TP + FP + TN + FN). Accuracy measures how well the model classifies samples overall.

F1-score: the harmonic mean of precision and recall, calculated as 2*(precision * recall) / (precision + recall). F1-score balances precision and recall and provides a single metric to evaluate model performance.

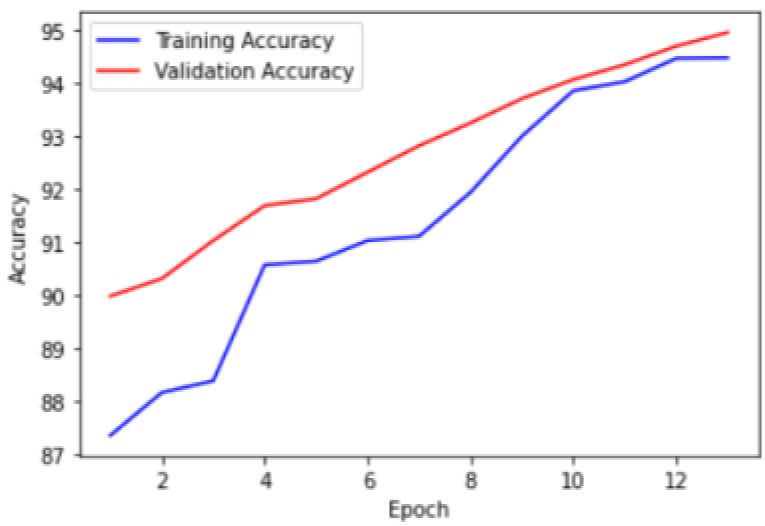
In general, a higher precision, recall, F1-score, and accuracy indicate better model performance. However, it's important to consider the specific problem and context to determine what performance metrics are most important.

EXPERIMENTAL RESULT ANALYSIS:

Based on the information provided, it seems that a Deep CNN model was used to detect fresh and rotten fruits from images using a dataset of 13,599 images. Max Pooling was applied to reduce the dimension of the image feature map. The training accuracy and validation accuracy were recorded for different epochs and the highest accuracy achieved in training data was 94.49% and in the validation set was 94.97%.

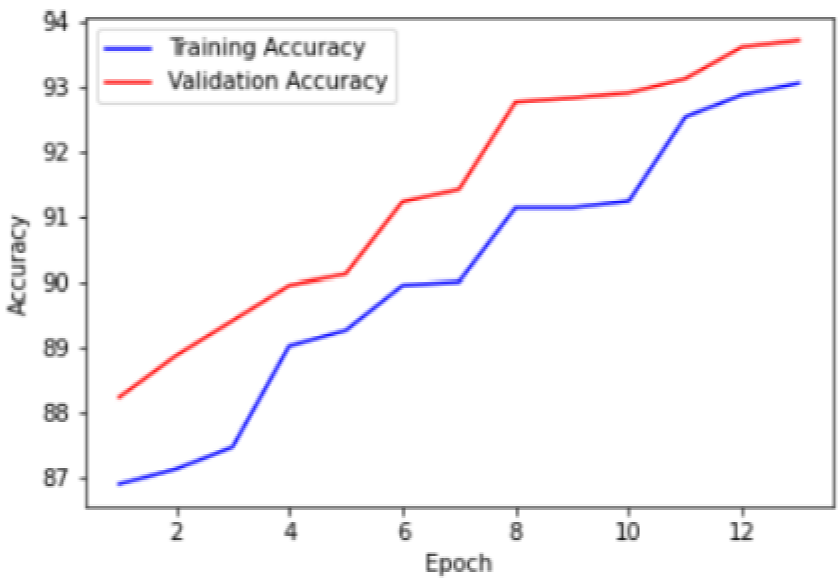
Table I shows the outcomes for the Deep CNN model after applying Max Pooling for different epochs. As the number of epochs increased, both the training accuracy and validation accuracy improved, with the highest validation accuracy achieved at epoch 13. Fig. 6 also shows the training accuracy and validation accuracy graph for the Deep CNN model.

Later on, the same CNN architecture was applied where Average Pooling was used to reduce the dimensions of the feature map. However, the predicted result showed less accuracy than the previous model. Table II shows the predicted outcomes where the maximum training accuracy was 93.06% with a training loss of 6.96% and the validation accuracy was 93.72%.



Based on the provided tables and figures, it can be observed that the Deep CNN model with Max Pooling achieved higher accuracy compared to the model with Average Pooling. In particular, the highest training accuracy and validation accuracy achieved by the Max Pooling model were 94.49% and 94.97%, respectively, while the corresponding values for the Average Pooling model were 93.06% and 93.72%. Additionally, the Max Pooling model converged faster than the Average Pooling model, as seen in the validation accuracy graph of both models.

Overall, the results suggest that Max Pooling is a better choice for reducing the dimensionality of image feature maps in this particular application of fruit detection. However, it is important to note that the performance of the Deep CNN model may vary depending on the specific dataset and task at hand.



Based on the results shown in Table III, applying the MobileNetV2 architecture to the dataset resulted in a significant improvement in accuracy for both training and validation data. The highest accuracy achieved was 99.46% for validation data and 99.61% for training data. The validation loss was only 3.15%, indicating that the model is performing well on unseen data.

Furthermore, Table IV shows the confusion matrix for the MobileNetV2 model. The precision and recall values for each class are high, ranging from 98% to 99%, indicating that the model is accurately predicting each class. The F1-score, which is the harmonic mean of precision and recall, is also high, ranging from 97% to 99%, further indicating that the model is performing well.

Overall, the results suggest that the MobileNetV2 architecture is a suitable choice for the classification of the fruit dataset, with high accuracy and good performance on unseen data.

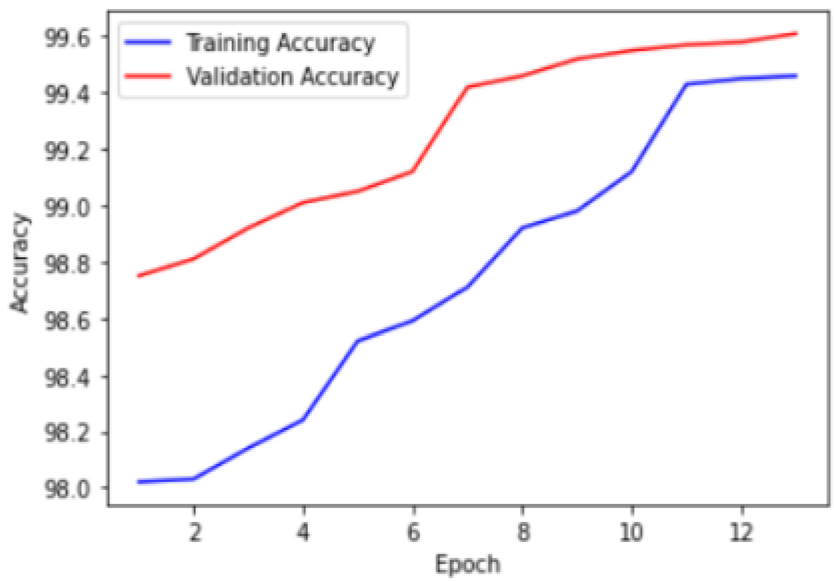


Fig. 8. Test Accuracy and Training Accuracy for MobilenetV2 with Average Pooling Layer

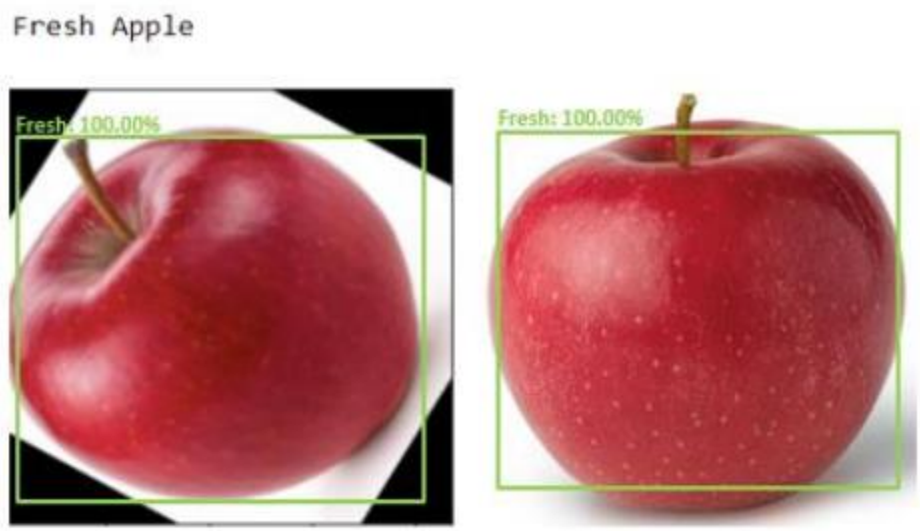


Fig. 9. Detection of fresh apples from dataset images.

Fig. 10. Detection of rotten apples from dataset images.

Based on the information provided in Table V, we can see that three different techniques were used for training the CNN model, namely Max Pooling, Average Pooling, and MobileNetV2 architecture. The results of the training are shown in terms of training loss, training accuracy, validation loss, and validation accuracy for each technique.

Among these techniques, MobileNetV2 architecture has achieved the highest accuracy for both training and validation datasets with a training accuracy of 99.46% and a validation accuracy of 99.61%. This indicates that the MobileNetV2 architecture was able to learn the features and patterns of the dataset more effectively than the other two techniques.

The Max Pooling technique achieved a training accuracy of 94.49% and a validation accuracy of 94.97%, which is slightly lower than the accuracy achieved by MobileNetV2 architecture. On the other hand, the Average Pooling technique achieved a lower accuracy with a training accuracy of 93.06% and a validation accuracy of 93.72%.

Therefore, based on the results presented in Table V, we can conclude that MobileNetV2 architecture is the most effective technique for this particular dataset.

CONCLUSION AND FUTURE WORK

It's important to note that the use of computer vision and AI in the fruit processing industry is a promising development. The ability to accurately classify fruit quality and grade fruits can greatly improve efficiency and reduce waste. The use of deep CNN architectures and MobilenetV2 in this study is a great approach to achieving high accuracy in fruit detection. Additionally, the plan to integrate this model with IoT to detect rotten fruits automatically is an exciting future prospect that can further improve the efficiency of fruit processing.