An AI-Based Approach to the Classification of Fruit Ripeness

Aratrika Pal

Irvington High School, 41800 Blacow Rd, Fremont, CA, 94538, USA; aratrikapal1@gmail.com
Mentor: Dr. Eric Sakk

ABSTRACT: Food waste is a serious issue worldwide, with a large percentage of fruits getting thrown away per year. Before the invention of technology, the ripeness of fruit was assessed through human labor, with workers spending several hours in the hot sun. However, this process was riddled with errors and was extremely inefficient, thereby cutting down the yields. Currently, methods for classifying multiple fruits and their level of ripening with the use of technology are lacking in quantity. The objective of this study is to design and develop an AI model based on deep learning techniques to classify the maturity levels of several types of fruits. The system consists of a model using the ResNet50 architecture, utilizing transfer learning from pre-trained weights. Data for this research were obtained through Kaggle. The fruit maturity level classification model achieved 91.95% accuracy, 92% precision, 92% recall, and 92% F1-score. According to the results, this system is able to effectively and accurately assess the maturity levels of various fruits, with very few false positives or false negatives. It allows for less wastage overall in the harvesting process.

KEYWORDS: Computer Science and Software Engineering, Artificial Intelligence, Transfer Learning, Resnet50.

Introduction

Food waste is a global problem, with around half of the fruits and vegetables produced worldwide being wasted annually.[1] A lot of this wastage is at the farm level, for a variety of reasons. The profitability of many fruits is very sensitive to the time that they're picked—produce that is picked at the wrong time ends up getting thrown out due to underripeness or overripeness, which contributes greatly to food wastage overall.

There are many changes in fruits that occur throughout the ripening process, including physical and chemical changes. Physical changes, which are observable to human workers, include changes in color, shape, size, and texture. Chemical changes are not as easily observable and require destructive processes done in labs. As a result, the main method that human laborers use today is detecting physical changes in produce.

Unfortunately, this raises another issue. Some farms cannot get enough laborers to harvest all the produce, and labor costs continue to rise, exacerbating the issue. Additionally, human error can lead to fruits getting picked that aren't in the appropriate maturity level for sale. Being able to identify a fruit's maturity level based on simply observing it takes skill and experience, and is prone to error, as it is not as reliable as destructive processes. For example, the harvesting of oil palm fruit bunches requires laborers with a lot of expertise in determining ripeness levels; unfortunately, the process is inefficient and filled with human error, leading to poor results and losses in the amount of yield.[2] The same is true for many other fruits. Additionally, long hours spent in the strong sunlight leave laborers prone to visual aberrations, which can cause the external color of fruit to appear differently and lead to improper ripeness determinations.

Recent data from Tesco and the World Wildlife Fund (WWF) indicates that around 40% of the world's food supply goes to waste, including on-farm losses.[3] These losses include fruit left unharvested or fruit harvested at the wrong time, though the exact percentage is difficult to estimate due to a lack of data.

All apples, bananas, and several other types of fruits are hand-picked by skilled workers, which requires a lot of manual labor. For example, apple farming labor costs account for up to 25% of total production costs, being one of the largest expenses for apple farmers. In general, labor is typically the highest cost of fruit production at 35-40% of annual gross revenue.[4] One of the main reasons for these high expenses is the need for experienced workers to identify fruit at the proper maturity level to pick. As a result, farms sometimes are not able to afford enough labor to harvest all the fruits, leading to waste.

Despite a growing interest in the use of AI for preventing food waste, there is a lack of research on detecting the ripeness stage of various fruits in one model. This research aims to design and develop a model capable of assessing the level of ripeness of 10 different classes of fruits (apples, bananas, dragonfruit, grapes, guavas, oranges, papayas, pineapples, pomegranates, and strawberries). Three levels of ripeness will be detected: unripe, ripe, and rotten.

Artificial Neural Networks (ANNs) are widely used in the food industry to solve the issues mentioned above. Many previous studies for detecting ripeness have only been trained for one or two fruits, due to a lack of existing data for training ANNs. These studies have used a non-destructive approach to identification, using analysis of RGB values and other similar techniques, as destructive processes are difficult to implement solely using AI.

In 2020, Harsawardana[5] *et al.* used a deep learning algorithm called a Convolutional Neural Network (CNN) to detect ripeness in oil palm fruit bunches by training the algorithm on

DOI: 10.36838/IJHSR89.12

several images of these fruit bunches. To account for a lack of data, they used an augmentation strategy called TenCrop to make 10 new images from each of the images in their original data set. In September 2020, Faisal[6] *et al.* also trained a CNN, this time to detect ripeness in dates. Additionally, Hadfi[7] and Yusoh identified the ripeness of bananas by getting the RGB values of images of bananas and utilizing fuzzy logic to represent degrees of truth. In this way, the model was able to not only detect underripe/ripe/overripe, but also percentages of ripeness and estimated shelf life of bananas.

Much later, in June 2023, Xiao[8] *et al.* used a Swin Transformer trained on images of apples and pears to detect ripeness. This new approach allows the model to capture fine details and overall context in an image using a "shifted window" approach, shifting window partitioning in alternate layers to reduce complexity and increase information flow. In November 2023, Tapia-Mendez[9] *et al.* conducted a study where two CNN-based models were used instead of one—the first model detected 32 types of fruits and vegetables, and the second model detected the maturity level of 6 of these types (apples, bananas, mangoes, oranges, potatoes, and tomatoes). This allowed for easier retraining and was one of the first studies to analyze the maturity levels of more than 1-2 types of fruits and vegetables.

Most recently, in June 2025, Han[2] *et al.* did another study on detecting ripeness in oil palm fruit bunches, this time using a Google Teachable Machine (GTM). A GTM is an online platform developed by Google to generate deep learning models without having to code them. It utilizes transfer learning (reusing a pre-existing model) to produce faster results. Overall, the majority of studies in this area have only been able to tackle the problem of detecting ripeness levels in 1-2 fruits, which is not very versatile.

The architecture used for the AI architecture in this study was the ResNet[10] model. Transfer learning was performed from the pre-trained weights of the model. These pre-trained weights were taken from ImageNet, and the initial weights of the first layers were frozen due to there being no necessity to retrain these layers. The system was trained on a set of 24,000 images of various fruits in multiple levels of maturity.

The proposed hypothesis is that an AI model can be developed to accurately discern the maturity levels of various fruits. The proposed system has a few limitations; the model is unable to accurately classify damaged, obstructed, or poorly illuminated fruits.

Methods

The development of the AI architecture was approached in multiple stages. The first stage was to collect an appropriate dataset for the problem at hand. Before beginning to train the model, the dataset had to be analyzed to ensure that there were no biases or limitations in the data available. This reduced the likelihood of receiving inaccurate results. Once this stage was done, the data had to be fed into the model so the algorithm could learn to predict outcomes based on new data. Finally, the model was tested on a smaller set of data to ensure accuracy in the results.

Dataset:

In order to develop an AI model to detect the ripeness of various fruits, a dataset containing several fruits and corresponding ripeness levels was required. The dataset used was obtained from Kaggle,[11] which is readily available secondary data. It contained 10 different fruits (apple, banana, dragonfruit, grape, guava, orange, papaya, pineapple, pomegranate, and strawberry). For each of these fruits, the dataset contained 3 stages of ripening (unripe, ripe, and rotten). There were 800 images per stage of ripening per fruit, leading to a total of 24,000 images in the dataset.

The dataset used for this study had not already been standardized for image format or resolution, so further preprocessing was required to bring it to a standardized format. All images were standardized to be 224x224, 300 ppi. Color channels were set to BGR, and all pixel values were divided by 255 to normalize the images. Setting color channels to BGR is crucial for model training, as the architecture being used was pretrained using the same color channels. Switching the color channels to use RGB or another format would interfere with the learning process and hinder model results. The analytical method used for the data is the 80-10-10 method, where 80% of the data is used for training, 10% for validation, and 10% for evaluation. That is, out of the total number of images, 19,200 images were used for training, 2,400 images were used for validation, and 2,400 images were used for evaluation. This image distribution was randomized for each test.

CNN Description:

The ResNet50 model was used for the AI architecture. This model consists of, as indicated by its name, 50 layers. 49 of these layers are convolutional layers, while one layer is a fully-connected layer. Since this model is a residual network, it increases accuracy through the use of 3-layer bottleneck residual mapping blocks.

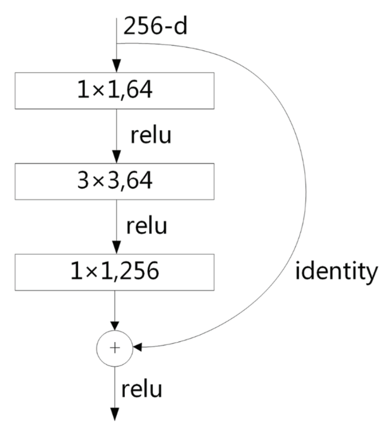

Figure 1: ResNet residual block input and output chart.

Figure 1 shows the structure of these residual blocks. The input to the block consists of 256 channels, meaning 256 features are being tracked in the input image. The first step, depicted by a rectangle, reduces the number of channels from 256 to 64. This is called dimensionality reduction, which reduces the number of channels to focus on the most important features in the input. After this, ReLU (Rectified Linear Unit) is activated, in which negative values are changed to zeros, and positive values are kept. This allows the model to learn more complex functions. Then, a spatial convolution is performed to detect local patterns and produce a feature map. ReLU is activated again, and then the image is restored to 256 channels. The original input is added to this, ReLU is activated one last time, and the final result is output.

These blocks remove the issue of vanishing gradient (gradients used to update the weights in earlier layers of a neural network becoming extremely small and hindering model progress) by setting up an alternate shortcut for the gradient to pass through. They also enable the model to learn an identity function, ensuring that the higher layers of the model do not perform any worse than the lower layers. This identity mapping bypasses the three convolutional layers described earlier to preserve the original signal.

The system was trained via transfer learning from the pre-trained Keras model. These pre-trained weights were taken from ImageNet.[12] It allows for models to be trained in shorter amounts of time, since the model does not have to train from scratch and already has some knowledge. Transfer learning applies the knowledge a model has learned from pre-training to improve generalization for another problem. In short, a model that was previously trained towards a certain task can be trained towards another task while retaining knowledge from the previous training, and this knowledge can be used to quicken pattern detection for the new task. Training a deep learning model from scratch could take up to days, but by utilizing transfer learning, this time can be reduced to a matter of hours while also increasing model accuracy.[13]

The Stochastic Gradient Descent (SGD) optimizer was utilized in the training of this model. The learning rate for this model was set to 0.01. Learning rate decay (reducing the learning rate after each update) was set to be small (1e-6) in order to gradually stabilize the model. Momentum was set to 0.9 to speed up model convergence and ensure that there are no erratic changes to model weights based on noisy gradients. The Nesterov Accelerated Gradient was enabled to improve the effects of momentum and increase accuracy in the process. SGD with momentum was specifically chosen for this research due to its stability with CNNs such as ResNet, allowing for better generalization during model learning compared to the use of other adaptive optimizers.

Performance Measures:

The original dataset had to be split into separate sets for training, validation, and testing as outlined above. Due to the large dataset being used, the split was not exact because of rounding errors; however, the difference was negligible (a 7-image difference out of 24,000), and this did not bias the final results in any way. Hyperparameters were modified using a grid search. Grid search is a common method for hyperparameter tuning. It trains the model with many hyperparameter values and then selects the model that provides the best performance on a validation set. Once the highest accuracy was reached during testing, the model was run for 15 epochs via a Kaggle notebook.

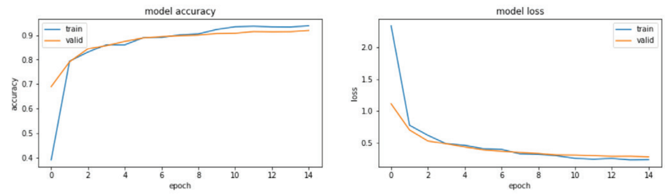

Figure 2: Training accuracy and loss graphs.

Figure 2 shows that over 15 epochs, accuracy increased drastically and loss decreased significantly. The model's accuracy on both training data and validation data consistently increased, staying at high values. Training and validation loss lines seem to decrease and converge at a low value as well. This shows that the model is well-fitted to the data, and is not overfitting or underfitting. The stopping criterion for this architecture was to be able to sustain an accuracy of over 90% while keeping loss values below 0.5. Based on these results, the final model was run for 15 epochs, as that amount satisfied these criteria.

Results and Discussion

Results:

After the model was finally trained, it was evaluated based on its accuracy when run on a testing dataset, which consisted of 10% of the original data. Below are the results:

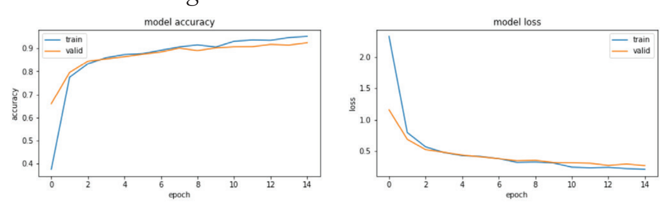

Figure 3: Final training accuracy and loss graphs.

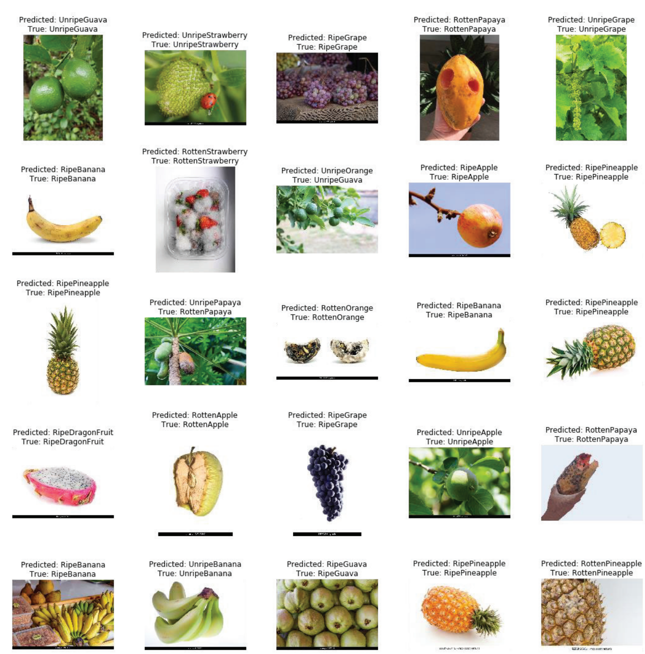

Figure 4: Model predictions.

From the final training and loss graphs as shown in Figure 3, it can be seen that training and validation lines have little to no variance. They are very far apart at the beginning and appear to

DOI: 10.36838/IJHSR89.12

converge after several epochs (when loss values go below 0.5), showing no evidence of overfitting or underfitting. Convergence occurs when the accuracy results have reached over 90%, displaying good model performance. Figure 4 shows the model's predictions for 25 random images from the testing dataset. The true values of the images are also displayed beneath the predictions. For a vast majority of images, the predictions are accurate.

No. Classes	Architecture	Epochs	Exec. Time	Acc.	P	R	F1
30	ResNet50	15	10690.809404	92.45%	93%	92%	92%

Figure 5: Final model metrics.

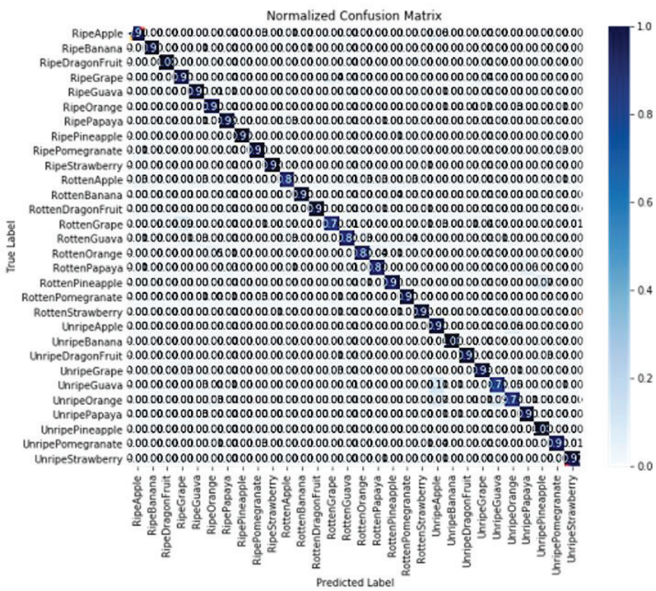

Figure 6: Final model confusion matrix.

Figure 5 shows the final metrics of the model, where execution time is recorded in seconds. As shown above, the model received 92.45% accuracy, 92% precision, 92% recall, and 92% F1 score. The accuracy of the model was 92.45%, which means that out of every 100 images, approximately 92 were correctly classified. The F1 score of the model is the harmonic mean of the model's precision and recall scores, and accounts for the model avoiding false positives and false negatives. Since the F1 score is high, it shows that the model is good at avoiding both false positives and false negatives. These results stayed consistent across 5 tests, where the tested images were randomized per test while keeping an even class distribution. Per test, a random distribution of images was chosen for the training, validation, and testing datasets. Across these tests, model metrics remained approximately the same, and accuracy remained high. Equations showing how to calculate these metrics are displayed below.

$$Accuracy: \frac{Correct\ Predictions}{Total\ Predictions}$$

$$Precision: \frac{Correct\ Predictions\ of\ a\ Class}{Total\ Predictions\ of\ the\ Class}$$

$$Recall: \frac{Correct\ Predictions\ of\ a\ Class}{All\ Images\ in\ the\ Class}$$

$$F1\ Score: \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

Figure 6 shows the confusion matrix for the model. A confusion matrix is a diagram with color-coded cells to show how often a model predicts a certain value for another true value. In the case of the above confusion matrix, a strong diagonal line can clearly be seen, showing that model predictions often match up with the true values of images. Looking at the confusion matrix, the model seemed to have the most difficulty with accurately predicting maturity levels for rotten grapes, unripe guavas, and unripe oranges, indicated by the blue squares with values of 0.7 instead of 0.8 or 0.9. Very few of the other squares have any color at all, showing a lack of false predictions. One thing to note with the confusion matrix is that since it is normalized, some false values have been rounded down and are not visible.

One issue that arose during this study was the training time for the model. Since ResNet50 is such a complex model, it takes much longer to train than a smaller CNN would. The final training time for all 15 epochs for the ResNet50 ended up totaling around 3 hours, and training had to be restarted several times due to the software timing out or the internet disconnecting during the training process. Overall, the process took several days.

Another unexpected problem that occurred was while actually obtaining the final model metrics and confusion matrix. Due to the shuffling done for the training process, the model was obtaining incorrect scores for precision, recall, and F1 score. To fix this, a copy of the validation set had to be made that was left unshuffled, so these values could be accurately calculated.

Discussion:

This study hypothesized that an AI model could be developed to accurately discern the maturity levels of various fruits. This hypothesis was correct, as demonstrated by the results above. The model has a high accuracy, with approximately 92 pictures out of every 100 being predicted correctly. It also has high precision and recall scores, showing that the model is unlikely to give false positives or negatives. In this context, it means that the model is balanced across the 10 types of fruits being observed. The high F1 score shows this as well, and overall, the metrics prove that the model is balanced and reliable. The confusion matrix supports this, showcasing that the model almost always predicts the correct level of maturity and type of fruit for an image, with rare errors.

This high model performance can be attributed to the nature of the model architecture itself, as well as the dataset used to train it. Due to using transfer learning to train the model on top of a large set of training data, the model had the data it needed to detect significant patterns in input images. These patterns were then used to accurately determine fruit maturity levels. The residual blocks within the structure of the ResNet50 architecture further this learning, allowing the model to focus on the most important patterns in input images without hindering model progress over time.

Conclusion

Food waste has been a very large problem in the world, and people around the world have consistently worked on solutions to it involving technologies such as AI. The creation of models such as the one in this study contributes to the solution. This study successfully identified the ripeness stages of 10 types of fruits using a dataset consisting of 24,000 images of the fruits in various maturity levels. The model was able to complete this task both accurately and quickly, much more so than the average human would be capable of doing.

The model in this study utilized transfer learning from the Keras ResNet50 model, and accuracy was enhanced through the fine-tuning of hyperparameters such as batch size and number of epochs. Overall, the model had high accuracy and F1 scores, showing that it was very reliable for detecting the maturity of all 10 fruits used (apple, banana, dragonfruit, grape, guava, orange, papaya, pineapple, pomegranate, and strawberry).

One big question that arose in this paper was the means of detecting the ripeness of fruits. While color/texture detection was the only main thing that could be done in this research, another major factor in determining fruit maturity is its softness. To put it into more quantifiable terms, the amount of pressure that one can put on a fruit without deforming it changes when maturity levels change.

Many fruits soften as they get riper, while others harden. Depending on the fruit, the amount of pressure one can place on it may increase or decrease throughout the ripening process. Unfortunately, there is no currently available data on this. Using implements such as a penetrometer to collect this data could help future studies in this field.[14] This model could even be implemented into robots to complete harvesting instead of human workers. This would further increase accuracy and efficiency, helping to solve the issue of food waste.

Another question that came up was the potential usage of other types of architectures. While this study made use of the ResNet50 model, future studies could seek to compare other types of architectures, such as VGG. Other architectures could potentially be more accurate than the one used in this study.

The usage of AI in solving problems such as food wastage remains a field to be explored in more depth. The technology used in this research could help increase accuracy and efficiency in the harvesting of many fruits, as well as minimize human labor in the process.

Acknowledgments

The author would like to thank Dr. Eric Sakk for mentorship throughout this research. The author attests that the ideas, graphics, and writing in this paper are entirely their own.

References

1. “How Many Fruits and Veggies Do We Waste Per Year?” Ruvi, 4 Jun. 2021, https://www.goruvi.com/blogs/news/644-million-tons-of-wasted-fruits-and-veggies-each-year.
2. Chong, Wei Han, Nor Azuana Ramli, Wan Mohd Rozaimi Wan Mustafa, Lilik Jamilatul Awalin. “Computer Vision and Artificial Intelligence (AI)-Based Ripeness Classification of Oil Palm Fruits in Oil Palm Plantations.” ASEAN Artificial Intelligence Journal, vol. 2, no. 1, Jun. 2025, pp. 44–58. karyailham.com.my, https://doi.org/10.37934/aaij.2.1.4458.
3. World Wide Fund for Nature (WWF) and Tesco. Driven to Waste: The Global Impact of Food Loss and Waste on Farms. Anthesis Ltd., 2021.
4. Advancements in Apple Picking: An Industry Addresses Tight Farm Labor Markets | Economic Research Service. https://www.ers.usda.gov/amber-waves/2023/june/advancements-in-apple-picking-an-industry-addresses-tight-farm-labor-markets. Accessed 11 Aug. 2025.
5. Harsawardana, Reza Rahutomo, Bharuno Mahesworo, Tjeng Wawan Cenggoro *et al.* “AI-Based Ripeness Grading for Oil Palm Fresh Fruit Bunch in Smart Crane Grabber.” IOP Conference Series: Earth and Environmental Science, vol. 426, no. 1, Feb. 2020, p. 012147. DOI.org (Crossref), https://doi.org/10.1088/1755-1315/426/1/012147.
6. Faisal, Mohammed, Mansour Alsulaiman, Mohammed Arafah, Mohamed Amine Mekhtiche *et al.* “IHDS: Intelligent Harvesting Decision System for Date Fruit Based on Maturity Stage Using Deep Learning and Computer Vision.” IEEE Access, vol. 8, 2020, pp. 167985–97. IEEE Xplore, https://doi.org/10.1109/ACCESS.2020.3023894.
7. Hadfi, Izzat Hafiz, and Zeratul Izzah Mohd Yusoh. “Banana Ripeness Detection and Servings Recommendation System Using Artificial Intelligence Techniques.” Journal of Telecommunication, Electronic and Computer Engineering (JTEC), vol. 10, no. 2–8, Jul. 2018, pp. 83–87. jtec.utem.edu.my, https://jtec.utem.edu.my/jtec/article/view/4464.
8. Xiao, Bingjie, Minh Nguyen, Wei Qi Yan. “Fruit Ripeness Identification Using Transformers.” Applied Intelligence, vol. 53, no. 19, Oct. 2023, pp. 22488–99. DOI.org (Crossref), https://doi.org/10.1007/s10489-023-04799-8.
9. Tapia-Méndez, Enoc, Irving A. Cruz-Albarran, Saul Tovar-Arriaga, Luis A. Morales-Hernández. “Deep Learning-Based Method for Classification and Ripeness Assessment of Fruits and Vegetables.” Applied Sciences, vol. 13, no. 22, Nov. 2023, p. 12504. DOI.org (Crossref), https://doi.org/10.3390/app132212504.
10. He, Kaiming, Xiangyu Zhang, Shaoqing Ren, Jian Sun *et al.* “Deep Residual Learning for Image Recognition.” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) [Las Vegas, NV, USA], 2016, pp. 770–78. DOI.org (Crossref), https://doi.org/10.1109/CVPR.2016.90.
11. Fruit Ripeness Level Dataset. https://www.kaggle.com/datasets/dudinurdiyansah/fruit-ripeness-level-dataset. Accessed 23 Aug. 2025.
12. Team, Keras. “Keras Documentation: ResNet and ResNetV2.” Keras.io, keras.io/api/applications/resnet/.
13. “What Is Transfer Learning? A Guide for Deep Learning.” Built In, https://builtin.com/data-science/transfer-learning. Accessed 12 Aug. 2025.
14. Pressure Testing | International Produce Training. 13 Mar. 2012, https://www.ipt.us.com/produce-inspection-resources/inspectors-blog/inspection-procedures/pressure-testing.

Author

Aratrika Pal, sophomore, Irvington High School, Fremont, CA. She designed/built a live-weather station called the Atmoprism. ACSEF 2nd Place. Robotics team leader, Gold medalist in MO. Teaches Math, debate & robotics to kids. 3rd in American Protege, solo performance in Carnegie Hall, NY. Gold Key, Scholastic Art and Writing Awards.

DOI: 10.36838/IJHSR89.12