Journal of Innovative Technology Convergence
Vol. 6, No. 3 September 2024, pp. 99-110
https://doi.org/10.69478/JITC2024v6n3a10

Classification of Kinalabasa Tomato Using Convolutional Neural Network

Jethel Mae G. Nalupano[1], Myline P. Omagap[2], Katherine Jane G. Fortaleza[3], Fema Rose B. Ecraela[4], John Jowil D. Orquia[5]*

Abstract: Determining the freshness of tomatoes is essential for evaluating their quality, impacting both consumer satisfaction and the economic benefits for farmers. Freshness is typically assessed by outer appearance, including color, size, and shape, with skin color indicating ripeness and influencing selling price. Properly assessing freshness also helps determine the shelf life of stored tomatoes. This study introduced an affordable and straightforward technique for classifying agricultural commodities using image processing, focusing on Kinalabasa tomatoes and classifying them based on color features into three categories. The study employed the MobileNetv2 architecture for training and testing the dataset, aiming to improve the classification of Kinalabasa tomatoes through deep learning techniques. The study evaluated the model's performance using metrics like accuracy, precision, sensitivity, and specificity. The optimal configuration for classifying Kinalabasa tomatoes was determined to be 10 epochs and a learning rate of 0.001. MobileNetv2 achieved a high accuracy rate of 94%, demonstrating its effectiveness in classifying tomatoes, though the specificity rate of 64% indicated some room for improvement in identifying negative instances. Comparing MobileNetv2 with ShuffleNet architectures will provide further insights into their respective effectiveness and performance in this classification task.

Keywords: Machine Vision, Convolutional Neural Network, Native Tomato Classification, Machine Learning

[1] College of Computer Studies, University of Antique, Sibalom, Antique, Western Visayas, Philippines
Email: nalupano.jethel@antiquespride.edu.ph
[2] College of Computer Studies, University of Antique, Sibalom, Antique, Western Visayas, Philippines
Email: omagap.myline@antiquespride.edu.ph
[3] College of Computer Studies, University of Antique, Sibalom, Antique, Western Visayas, Philippines
Email: fortaleza.katherinejane@antiquespride.edu.ph
[4] College of Computer Studies, University of Antique, Sibalom, Antique, Western Visayas, Philippines
Email: femarose.ecraela@antiquespride.edu.ph
[5]* College of Computer Studies, University of Antique, Sibalom, Antique, Western Visayas, Philippines
Email: jjorquia@antiquespride.edu.ph (Corresponding Author)

Received [April 2, 2024]; Revised [June 28, 2024]; Accepted [July 31, 2024]

Published by InnoCon Publishing
ISSN 2704-4440

1. Introduction

Despite increased industrialization, the Philippines remained predominantly agricultural, with crop production reaching a gross value of PhP 429.7 billion in the first quarter of 2019 [1]. Tomatoes (Solanum lycopersicum), known locally as "Kamatis," were highly valued for their medicinal benefits, including reducing the risk of cancer, cardiovascular disease, and osteoporosis, and their nutritional content, such as phosphorus, iron, calcium, and vitamins C and D [2]. Lycopene, the primary carotenoid in tomatoes, accumulated mainly during the fruit's final ripening stage, accounting for 80% of its carotenoid content [3]. The assessment of tomato freshness, indicated by outer characteristics like color, size, and shape, was crucial for determining quality and influencing selling prices.

Tomatoes thrived in high temperatures and brightness, but their productivity could be limited by fruit-bearing tendencies and the ability to protect them from damage. High-quality yields were economically beneficial for farmers, as better produce fetched higher prices and was used in various food products such as jams, sauces, and tinned goods. Determining the ripeness stages of tomatoes also helped establish their shelf life if stored [4]. Accurate ripeness assessment ensured that both consumers and farmers benefited from high-quality, nutritious fruits.

This study proposed a technique for classifying Kinalabasa tomatoes using Convolutional Neural Networks (CNNs), aiming for cost-effective and efficient classification inspections of agricultural commodities. Utilizing image processing, the setup processed tomato images to classify them into three categories based on color features. CNNs, known for their pattern recognition abilities, involved multiple layers, including input, activation function, pooling, and fully connected layers, to analyze images. The study aimed to compare the effectiveness of MobileNetv2 and ShuffleNet architectures in classifying tomatoes and optimizing learning rates and epochs to achieve the best results [5-7].

2. Related Works

In the rapidly evolving field of agriculture technology, this study explores advanced deep learning techniques, particularly convolutional neural networks (CNNs), to enhance the automation and accuracy of tomato ripeness assessment and fruit classification. Convolutional neural networks (CNNs) were extensively used for image processing, classification, segmentation, and other data processing tasks. As described by Thomas, CNNs are deep learning systems capable of recognizing differences between items in an image by learning their importance and properties, such as learnable weights and biases [8][9]. These networks adaptively learned spatial hierarchies of information through various building elements, including convolutional, pooling, and fully connected layers [10]. The performance of CNN models was evaluated using forward propagation on training datasets, while backpropagation with gradient descent optimization was used to update learnable parameters like kernels and weights.

Deep learning achieved significant success across various application domains. Its rapid expansion in fields such as image processing, computer vision, speech recognition, machine translation, and medical imaging [11]. These applications often utilized supervised, semi-supervised, and unsupervised learning methodologies. Compared to traditional machine learning methods, deep learning demonstrated state-of-the-art performance in numerous areas, including bioinformatics, natural language processing, and cybersecurity.

MobileNetv2, a CNN designed for mobile applications, aimed to reduce model size and complexity while maintaining efficiency. MobileNetv2 employed depth-wise separable convolution as effective building blocks and featured shortcut connections between bottlenecks and linear bottlenecks between layers [12]. This architecture allowed the model to transition from lower-level concepts like pixels to higher-level descriptors like image categories, with bottlenecks encoding intermediate inputs and outputs. These features, combined with shortcut connections, facilitated faster training and improved accuracy, making MobileNetv2 an effective mobile-oriented network.

ShuffleNet simplified processing by employing channel shuffle to move data between feature channels and pointwise group convolution. Designed for portable computing devices, ShuffleNet provided accurate calculations at an affordable price [13]. The channel shuffle operation enabled cross-channel information flow between multiple convolution layers, consistently improving classification results with minimal speed penalties. This technique effectively increased classification accuracy in various settings while maintaining efficiency.

The synthesis of related works revealed that various studies had successfully employed deep learning techniques, particularly convolutional neural networks (CNNs), to enhance the automation and accuracy of tomato ripeness assessment and fruit classification. Researchers demonstrated the effectiveness of CNN-based systems for automated tomato classification, highlighting their ability to reduce errors associated with manual methods. Some addressed limitations in previous approaches by presenting a CNN model that distinguished fresh and rotten fruits, using advanced systems with neural networks and transfer learning to achieve state-of-the-art accuracy in sorting tomatoes based on maturity. These studies collectively underscored the potential of deep learning models to significantly improve efficiency and precision in agricultural processes, thereby reducing food waste and optimizing production.

3. Methodology

The methodology employed in this study was to utilize a pre-trained deep learning model to classify three divided classes of Kinalabasa tomatoes. An experimental approach was used to determine the classification accuracy, precision, sensitivity, and specificity of the proposed deep learning models, such as MobileNetv2, compared to ShuffleNet architecture. The researchers utilized the MobileNetv2 architecture for the training and testing of the dataset. In this study, Kinalabasa tomato samples were divided into three classes: ripe, unripe, and overripe. The research used 3,000 tomato samples overall, with 1,000 for each freshness stage. MobileNetv2 was employed to differentiate the three classes and determine its performance based on accuracy, precision, sensitivity, and specificity. After determining MobileNetv2's performance, it was compared to ShuffleNet to assess the significance of their differences.

3.1 Dataset

The study used Kinalabasa tomato samples classified as ripe, unripe, and overripe, as shown in Table 1. The researchers generated their own dataset by collecting 3,000 images of Kinalabasa tomatoes found locally in the Philippines, manually capturing the sample images indoors with ring lights and mobile phones in high-resolution JPG format. The MobileNetv2 model source codes were obtained from Python. Deep Learning Toolbox software and the MobileNetv2 architecture were applied during training [14].

Table 1. Sample Images of Kinalabasa Tomato

Unripe	Ripe	Overripe
		
		
		

3.2 Hyperparameter Optimization

Hyperparameters were crucial for model training, affecting training time, resource requirements, convergence, and accuracy. The optimization of hyperparameters was done through manual search, testing combinations of hyperparameter values, and training the model for each combination to find the best result [15]. The researchers focused on fine-tuning the learning rate, which controls model weight changes in response to error, and the epoch, indicating the number of passes of the entire training dataset completed by the machine learning algorithm [16][17]. The data was split into 80% training and 20% testing to handle small sample sizes using augmented data.

Figure 1. The Camera Settings Used During Image Acquisition

Sample images shown in Table 1 were manually captured indoors using an iPhone 8 Plus mobile phone equipped with a 12-megapixel camera positioned 5 cm above a white bond paper surface. The photos were taken in natural light, rotated and flipped, and shot in 2 – 5 frames in a steady state. A total of 1,000 images were taken and saved in folders in JPG format. The camera settings used during the image acquisition were: image size 3024 x 4032, no flash, image type JPG, and aperture f/1.8, as shown in Figure 1.

3.3 Data Analysis Procedure

The performance of the MobileNetv2 model was evaluated using accuracy, precision, sensitivity, and specificity, derived from confusion matrices [18]. These measures determined if the MobileNetv2 model was effective in classifying the freshness of Kinalabasa tomatoes. A two-way ANOVA test was used to compare the performance of MobileNetv2 with ShuffleNet based on freshness classifications. This statistical tool evaluated the influence of two independent variables on one dependent variable and was used to confirm if there was a significant performance difference between the proposed deep learning model and existing architectures.

4. Results and Discussion

A dataset of Kinalabasa tomatoeswas collected by capturing images specifically focused on these tomatoes and removing the background to isolate the object of interest. This image processing task was performed using various background removal techniques in Google Colab (Python). Since the original images were in HEIC format, we converted them to the more commonly used PNG format using an HEIC converter. By analyzing the color distribution within the tomato images, we categorized the tomatoes into three groups: ripe, unripe, and overripe. These preprocessing steps were necessary for subsequent analysis or machine learning tasks involving image datasets. The MobileNetv2 architecture was configured with hyperparameters such as learning rate and epoch, testing several values to determine the best settings for classifying Kinalabasa tomatoes based on accuracy, precision, sensitivity, and specificity. To prepare the dataset for training models with MobileNetv2 and ShuffleNet architectures, we organized labeled images of tomatoes into separate folders, set up Google Colab, uploaded the dataset using unzip, and installed necessary libraries and frameworks like TensorFlow using pip install. Appropriate data loading and preprocessing techniques were then applied for the framework.

Table 2. Testing Results of MobileNetv2 Architecture Based on Accuracy, Precision, Sensitivity, and Specificity

Learning Rate	Epoch	Accuracy	Precision	Sensitivity	Specificity
0.01	5	0.90	0.90	0.90	0.60
0.001	5	0.91	0.91	0.91	0.62
0.0001	5	0.91	0.91	0.91	0.59
0.00001	5	0.91	0.91	0.91	0.52

0.01	10	0.92	0.92	0.92	0.63
0.001	10	0.94	0.94	0.94	0.64
0.0001	10	0.93	0.93	0.93	0.50
0.00001	10	0.93	0.93	0.93	0.51

The MobileNetv2 architecture shown in Table 2 was successfully configured, and the values that gave the best results for Kinalabasa tomato were 10 for the epoch and 0.001 for the learning rate. The MobileNetv2 architecture achieved a 94% accuracy rate, 94% precision rate, 94% sensitivity rate, and 64% specificity rate. The following table shows the results based on accuracy, precision, sensitivity, and specificity of the pre-trained deep learning models tested with different values of learning rate and epoch.

Table 3. Testing Results of ShuffleNet Architecture Based on Accuracy, Precision, Sensitivity, and Specificity

Learning Rate	Epoch	Accuracy	Precision	Sensitivity	Specificity
0.01	5	0.91	0.91	0.91	0.54
0.001	5	0.90	0.90	0.90	0.43
0.0001	5	0.91	0.91	0.91	0.56
0.00001	5	0.89	0.89	0.89	0.57
0.01	10	0.91	0.91	0.91	0.61
0.001	10	0.93	0.93	0.93	0.54
0.0001	10	0.92	0.92	0.92	0.50
0.00001	10	0.93	0.93	0.93	0.53

The ShuffleNet architecture in Table 3 was successfully configured, and the values that gave the best results for Kinalabasa tomato were 10 for the epoch and 0.001 for the learning rate. The ShuffleNet architecture achieved a 54% accuracy rate, 93% precision rate, 93% sensitivity rate, and 64% specificity rate. The following table shows the results based on accuracy, precision, sensitivity, and specificity of the pre-trained deep learning models tested with different values of learning rate and epoch.

The testing results shown in Table 4 of the MobileNetv2 architecture for classifying Kinalabasa tomatoes revealed that a learning rate of 0.001 and 10 epochs consistently provided the highest performance across all metrics for ripe, unripe, and overripe categories. This configuration achieved the best accuracy (0.94), with precision ranging from 0.90 to 0.96, sensitivity from 0.86 to 1.00, and specificity of 0.64. Despite the high accuracy, precision, and sensitivity, the specificity remained relatively lower, indicating some challenges in correctly identifying negative instances.

Table 4. Testing Results of MobileNetv2 Architecture Based on Accuracy, Precision, Sensitivity, and Specificity of Ripe, Unripe, and Overripe Tomato Classes

Testing Results of MobileNetv2 Architecture													
		Ripe Tomato				Unripe Tomato				Overripe Tomato			
Learning Rate	Epoch	Accuracy	Precision	Sensitivity	Specificity	Accuracy	Precision	Sensitivity	Specificity	Accuracy	Precision	Sensitivity	Specificity
0.01	5	0.90	0.87	0.91	0.60	0.90	0.91	0.80	0.60	0.90	0.91	0.99	0.60
0.001	5	0.91	0.93	0.84	0.62	0.91	0.86	0.90	0.62	0.91	0.95	1.00	0.62
0.0001	5	0.91	0.88	0.91	0.59	0.91	0.91	0.84	0.59	0.91	0.95	0.99	0.59
0.00001	5	0.91	0.86	0.89	0.52	0.91	0.90	0.84	0.52	0.91	0.96	0.97	0.52
0.01	10	0.92	0.90	0.93	0.63	0.92	0.93	0.85	0.63	0.92	0.94	1.00	0.63
0.001	10	0.94	0.90	0.96	0.64	0.94	0.96	0.86	0.64	0.94	0.95	1.00	0.64
0.0001	10	0.93	0.90	0.93	0.50	0.93	0.91	0.89	0.50	0.93	0.97	0.97	0.50
0.00001	10	0.93	0.87	0.95	0.51	0.93	0.94	0.87	0.51	0.93	0.98	0.98	0.51

Table 5. Testing Results of ShuffleNet Architecture Based on Accuracy, Precision, Sensitivity, and Specificity of Ripe, Unripe, and Overripe Tomato Classes

Testing Results of ShuffleNet Architecture													
		Ripe Tomato				Unripe Tomato				Overripe Tomato			
Learning Rate	Epoch	Accuracy	Precision	Sensitivity	Specificity	Accuracy	Precision	Sensitivity	Specificity	Accuracy	Precision	Sensitivity	Specificity
0.01	5	0.91	0.92	0.86	0.54	0.91	0.86	0.88	0.54	0.91	0.95	0.98	0.54
0.001	5	0.90	0.92	0.83	0.43	0.90	0.83	0.92	0.43	0.90	0.98	0.95	0.43
0.0001	5	0.91	0.86	0.92	0.56	0.91	0.90	0.82	0.56	0.91	0.95	0.98	0.56
0.00001	5	0.89	0.82	0.91	0.57	0.89	0.89	0.78	0.57	0.89	0.95	0.98	0.57
0.01	10	0.91	0.88	0.89	0.61	0.91	0.90	0.85	0.61	0.91	0.95	0.99	0.61
0.001	10	0.93	0.86	0.96	0.54	0.93	0.94	0.85	0.54	0.93	0.98	0.98	0.54
0.0001	10	0.92	0.88	0.90	0.50	0.92	0.89	0.88	0.50	0.92	0.98	0.98	0.50
0.00001	10	0.93	0.91	0.91	0.53	0.93	0.91	0.89	0.53	0.93	0.96	0.98	0.53

The testing results in Table 5 of the ShuffleNet architecture showed that a learning rate of 0.001 and 10 epochs yielded the highest performance across the three tomato classifications: ripe, unripe, and overripe. This configuration achieved the best accuracy (0.93), with precision ranging from 0.86 to 0.91, sensitivity from 0.85 to 0.98, and specificity from 0.50 to 0.61. Although ShuffleNet demonstrated high sensitivity and precision, its specificity was generally lower compared to MobileNetv2, suggesting that it struggled more with correctly identifying negative instances.

Table 6. Accuracy, Precision, Sensitivity, and Specificity (Architecture)

Architecture	Accuracy	Precision	Sensitivity	Specificity
MobileNetv2	0.9188	0.9183	0.9196	0.5762
ShuffleNet	0.9125	0.9112	0.9113	0.5350

Both architectures demonstrated high accuracy values, as shown in Table 6, as determined by a two-way ANOVA, indicating their capability to make correct predictions for a significant portion of the dataset. MobileNetv2 achieved a slightly higher accuracy of 0.9188 compared to ShuffleNet's accuracy of 0.9125. The results of the comparison provided insights into the overall correctness, precision, sensitivity, and specificity of the models' predictions, with MobileNetv2 generally performing better across these metrics. The two-way ANOVA analysis revealed no statistically significant differences between the performances of MobileNetv2 and ShuffleNet. While both architectures exhibited high accuracy, MobileNetv2 showed superior precision, sensitivity, and specificity, suggesting a better balance in identifying both positive and negative samples. ShuffleNet, on the other hand, had lower specificity, indicating a higher rate of false positives. This study utilized MobileNetv2 and ShuffleNet to classify Kinalabasa tomatoes into three categories: ripe, unripe, and overripe, highlighting the importance of considering specific requirements and trade-offs when selecting between these architectures.

Table 7. Test of Between-Subjects Effect with Accuracy as Dependent Variable

Source	SS	df	MS	F	p	ES	Source	SS	df	MS	F	P	ES
Architecture	0	1	0	8.333	0.006	0.159	Architecture	0	1	0	2.66	0.111	0.062
Epoch	0.005	1	0.005	96.333	0	0.686	Learning Rate	0.001	3	0	1.241	0.308	0.085
Architecture * Epoch	1.88E-05	1	1.88E-05	0.333	0.567	0.008	Architecture * Learning Rate	0	3	6.88E-05	0.39	0.761	0.028
Error	0.002	44	5.63E-05				Error	0.007	40	0			
Total	40.25	48					Total	40.25	48				

The analysis in Table 7 showed that the main effect of epoch on accuracy was highly significant, with an SS of 0.005, df of 1, MS of 0.005, an F-value of 96.333, and a p-value of 0.000, indicating a highly significant difference in accuracy between different epoch conditions and a large effect size of 0.686. However, the interaction between architecture and epoch did not significantly affect accuracy, as indicated by an F-value of 0.333, a p-value of 0.567, and a very small effect size of 0.008. A two-way ANOVA was conducted to classify the freshness of Kinalabasa tomatoes, and the results presented also indicate no significant difference in accuracy due to the factors of architecture, learning rate, or their interaction. The main effect of architecture on accuracy was not statistically significant, with an SS of 0.000, df of 1, MS of 0.000, an F-value of 2.660, and a p-value of 0.111, suggesting that the observed difference in accuracy between different architectural conditions was likely due to chance, with a small effect size of 0.062.

Table 8. Test of Between-Subjects Effect with Precision as Dependent Variable

Source	SS	df	MS	F	P	ES	Source	SS	df	MS	F	P	ES
Architecture	0.001	1	0.001	0.36	0.551	0.008	Architecture	0.001	1	0.001	0.313	0.579	0.008
Epoch	0.005	1	0.005	2.753	0.104	0.059	Learning Rate	0.001	3	0	0.157	0.925	0.012
Architecture * Epoch	5.21E-05	1	5.21E-05	0.031	0.861	0.001	Architecture * Learning Rate	0	3	7.99E-05	0.041	0.989	0.003
Error	0.074	44	0.002				Error	0.077	40	0.002			
Total	40.247	48					Total	40.247	48				

The two-way ANOVA conducted to classify the freshness of Kinalabasa tomatoes showed no significant difference in the dependent variable (precision) due to the factors of architecture, epoch, learning rate, or their interactions, as presented in Table 8. For architecture, the F-value was 0.360 with a p-value of 0.551, indicating that the observed differences in precision were likely due to chance, with an effect size (ES) of 0.008, suggesting a very small effect. Similarly, for architecture in relation to learning rate, the F-value was 0.313 with a p-value of 0.579, also indicating that the differences were likely due to chance, with an ES of 0.008. These results indicated that the variability in precision could not be attributed to the different architectural conditions tested, and the differences observed were not statistically significant.

Another two-way ANOVA conducted to classify the freshness of Kinalabasa tomatoes showed no significant difference in sensitivity due to the factors of architecture, epoch, or their interaction. For architecture, the SS was 0.001, the F-value was 0.223, and the p-value was 0.639, suggesting that differences in sensitivity were likely due to chance with an effect size (ES) of 0.005. Similarly, for learning rate, the F-value was 0.059 and the p-value was 0.981, indicating no significant impact on sensitivity. The interaction effect between architecture and learning rate also had an SS of 6.667E-005, an F-ratio of 0.005, and a p-value of 0.999, showing no significant effect on sensitivity. However, a significant difference in specificity based on architecture, with an SS of 0.020, an F-ratio of 7.186, and a p-value of 0.010, suggests a significant impact. For Epoch, the p-value was 0.809, indicating no significant impact on specificity. The interaction effect between Architecture and Epoch had an SS of 0.003, an F-ratio of 1.115, and a p-value of 0.297, showing no significant effect on specificity. It also revealed a significant difference in specificity based on architecture, with an F-ratio of 17.177 and a p-value of 0.000 and learning rate, with an F-ratio of 8.134 and a p-value of 0.000, indicating significant impacts from both factors.

5. Conclusion and Recommendations

In creating our dataset for classifying Kinalabasa tomatoes, we collected and annotated a diverse set of images, split the dataset for training and evaluation, preprocessed the data, trained a model using machine learning techniques, evaluated its performance, and iterated on the process to achieve better results. The MobileNetv2 architecture was successfully configured, with the optimal values for epoch and learning rate determined to be 10 and 0.001, respectively. MobileNetv2 achieved a 94% accuracy rate, 94% precision rate, 94% sensitivity rate, and 64% specificity rate. These results indicate that the MobileNetv2 architecture effectively classified the images in the Kinalabasa Tomato Dataset, performing well across all performance metrics.

The formulated hypotheses aimed to explore and compare different performance metrics between two architectures, MobileNetv2 and ShuffleNet. The first hypothesis stated that there was no significant difference in the accuracy rates between MobileNetv2 and ShuffleNet, suggesting that both architectures performed equally well in terms of overall accuracy. The second hypothesis stated that there was no significant difference in the precision rates, indicating similar precision rates for both architectures. The third hypothesis proposed no significant difference in sensitivity rates, implying both architectures had similar abilities to identify positive instances correctly. The fourth hypothesis stated a significant difference in specificity rates between MobileNetv2 and ShuffleNet, suggesting a notable distinction in their abilities to correctly identify negative instances.

In the field of agriculture, seeking guidance from agricultural experts, local farmers, or horticultural organizations specializing in tomatoes or native varieties was crucial. They provided valuable insights into the classification of the Kinalabasa tomato and similar local varieties. Collaborative efforts with other farmers, researchers, and organizations interested in preserving native varieties or studying local biodiversity were also recommended. By sharing information and experiences, collective contributions were made towards the classification and conservation of the Kinalabasa tomato.

References

[1] E. O. Brown, R. V. Ebora, and F. L. C. Decena, "*The Current State, Challenges and Plans for Philippine Agriculture*," FFTC Agricultural Policy Platform (FFTC-AP), November 2018, https://ap.fftc.org.tw/article/500.

[2] D. Bhowmik, K. P. S. Kumar, S. Paswan, and S. Srivastava, "*Tomato - A Natural Medicine and Its Health Benefits*," Journal of Pharmacognosy and Phytochemistry, vol. 1, no. 1, May 2012, pp. 33-43, ISSN 2278-4136.

[3] W. Klunklin and G. Savage, "*Effect on Quality Characteristics of Tomatoes Grown Under Well-Watered and Drought Stress Conditions*," Foods, vol. 6, no. 8, August 2017, https://doi.org/10.3390/foods6080056.

[4] M. B. Deshetti, M. Y. Teggi, and A. Durgad, "*Growth and Export Performance of Tomato in India*," International Journal of Economic and Business Review, vol. 3, no. 1, January 2015, pp. 48-52, e-ISSN: 2347-9671.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "*Imagenet Classification with Deep Convolutional Neural Networks*," Communications of the ACM, vol. 60, no. 6, May 2017, pp. 84-90, https://doi.org/10.1145/306538.

[6] M. Brahimi, M. Arsenovic, S. Laraba, S. Sladojevic, K. Boukhalfa, and A. Moussaoui, "*Deep Learning for Plant Diseases: Detection and Saliency Map Visualisation*," in *Human and Machine Learning. Human–Computer Interaction Series*, J. Zhou, F. Chen (Eds.), Springer International Publishing: Cham, Switzerland, June 2018, pp. 93-117, https://doi.org/10.1007/978-3-319-90403-0_6.

[7] K. P. Ferentinos, "*Deep Learning Models for Plant Disease Detection and Diagnosis*," Computers and Electronics in Agriculture, vol. 145, February 2018, pp. 311-318, https://doi.org/10.1016/j.compag.2018.01.009.

[8] C. Thomas, "*An introduction to Convolutional Neural Networks*," Medium, May 2019, https://towardsdatascience.com/an-introduction-to-convolutional-neural-networks-eb0b60b58fd7.

[9] S. Saha, "*A Comprehensive Guide to Convolutional Neural Networks – the ELI5 Way*," Medium, December 2018, https://saturncloud.io/blog/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way/.

[10] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "*Convolutional Neural Networks: An Overview and Application in Radiology*," Insights into Imaging," vol. 9, June 2018, pp. 611-629, https://doi.org/10.1007/s13244-018-0639-9.

[11] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, M. Hasan, B. C. Van Essen, A. A. S. Awwal, V. K. Asari, "*A State-of-the-Art Survey on Deep Learning Theory and Architectures*," Electronics, vol. 8, no. 3, March 2019, pp. 292, https://doi.org/10.3390/electronics8030292.

[12] M. Yawar, "*MobileNet*," Code Studio, March 2024, https://www.naukri.com/code360/library/mobilenet.

[13] V. Bhole and A. Kumar, "*A Transfer Learning-based Approach to Predict the Shelf life of Fruit*," Inteligencia Artificial, vol. 24, no. 67, June 2021, pp. 102-120, https://doi.org/10.4114/intartif.vol24iss67pp102-120.

[14] P. Kasture and K. Shirsath, "*Enhancing Stock Market Prediction: A Hybrid RNN-LSTM Framework with Sentiment Analysis*," Indian Journal of Science and Technology, vol. 17, no. 18, April 2024, pp. 1880-1888, https://doi.org/10.17485/IJST/v17i18.466.

[15] S. S. Pandi, V. R. Chiranjeevi, T. Kumaragurubaran, and P. Kumar, "*Improvement of Classification Accuracy in Machine Learning Algorithm by Hyper-Parameter Optimization*," in Proc. 2023 International Conference on Research Methodologies in Knowledge Management, Artificial Intelligence and Telecommunication Engineering (RMKMATE), Chennai, India, November 1-2, 2023, pp. 1-5, https://doi.org/10.1109/RMKMATE59243.2023.10369177.

[16] J. Brownlee, "*Difference Between a Batch and an Epoch in a Neural Network*," Machine Learning Mastery, August 2022, https://machinelearningmastery.com/difference-between-a-batch-and-an-epoch/.

[17] S. J. Bell, O. P. Kampman, J. Dodge, and N. D. Lawrence, "*Modeling the Machine Learning Multiverse*," in Proc. of the 36th International Conference on Neural Information Processing System, June 2022, pp. 18416-18429, https://doi.org/10.48550/arXiv.2206.05985.

[18] D. Bertsimas, D. Zhuo, J. Dunn, J. Levine, E. Zuccarelli, N. Smyrnakis, Z. Tobota, B. Maruszewski, J. Fragata, G. E. Sarris, "*Adverse Outcomes Prediction for Congenital Heart Surgery: A Machine Learning Approach*," World Journal for Pediatric and Congenital Heart Surgery, vol. 12, no. 4, July 2021, pp. 453-460, https://doi.org/10.1177/21501351211007106.