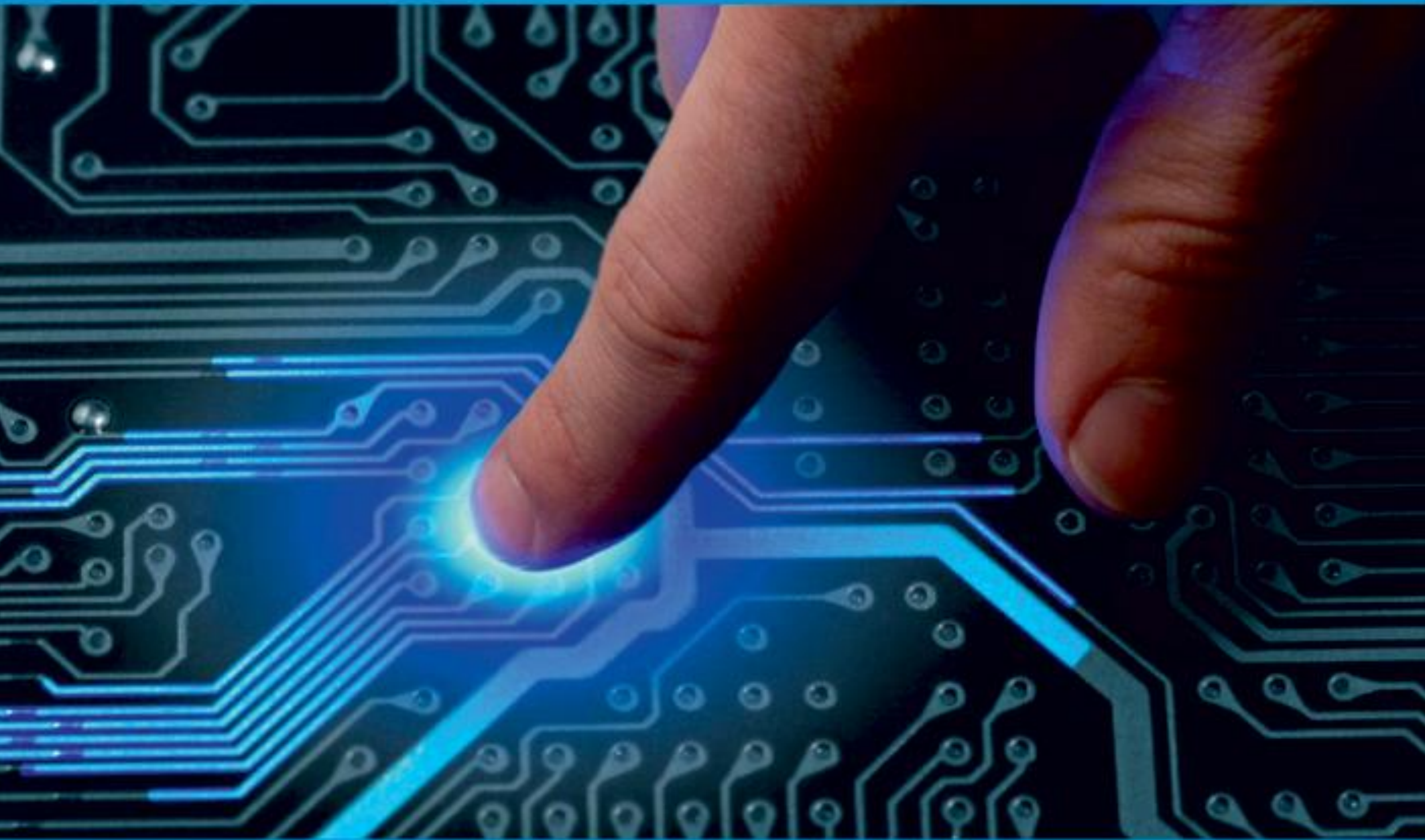
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 12, Special Issue 1, March 2024

1st International Conference on Machine Learning, Optimization and Data Science

Organized by

Department of Computer Science and Engineering, Baderia Global Institute of Engineering and Management, Jabalpur, India

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.379

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || Monthly Peer Reviewed & Refereed Journal |

| "1st International Conference on Machine Learning, Optimization and Data Science"|

Organized by

| Department of Computer Science and Engineering, Baderia Global Institute of Engineering and Management, Jabalpur|

|| Volume 12, Special Issue 1, March 2024 ||

Investigating Optimization Methods and Loss Functions for Training Neural Networks: A Comparative Study

Mallika Dwivedi, Divya Pandey, Jaya Choubey

Assistant Professor, Dept. of C.S.E., RGPV University Bhopal, India

Assistant Professor, Dept. of C.S.E., RGPV University Bhopal, India

Assistant Professor, Dept. of C.S.E., RGPV University Bhopal, India

ABSTRACT: In training neural networks for plant disease prediction, this study examines the efficacy of several optimization techniques and loss functions. Various combinations of optimizers and loss functions are investigated to assess their effect on model performance using a comparative research technique. The study makes use of a dataset that includes pictures of both healthy and diseased plants together with labels indicating whether or not the plants are diseased. Plant disease prediction models can be made more accurate by employing the best optimization strategies and loss functions, which are discovered through extensive testing and research. The results further our knowledge of the best practices for neural network training in agricultural applications, especially with regard to disease control and plant health monitoring.

KEYWORDS: Neural networks, optimization methods, loss functions, plant disease prediction, comparative analysis, agricultural applications.

I. INTRODUCTION

Pests and plant diseases have a major effect on plant output worldwide, resulting in losses for the economy and society, especially in unfavorable environmental circumstances. The application of pesticides, including fungicides, insecticides, and herbicides, is a major component of current management techniques. Early identification and classification of plant diseases become essential for improving food security in light of the United Nations' ambitious 2030 objective of reaching zero hunger. With a projected 7.837 billion people on the planet as of 2021, there will be a growing need for food. A priority of effective food production is necessary to combat global hunger, as the number of hungry people is expected to climb alarmingly to 828 million by 2021.

In order to control pests and plant diseases, farmers have historically turned to chemical remedies like herbicides and insecticides, which have temporarily increased crop yields. This study provides a comprehensive analysis of the most recent approaches to disease identification and Plant Disease Detection (PDD), with a focus on the widespread application of Artificial Intelligence (AI), encompassing Machine Learning (ML), Deep Learning (DL), and Image Processing (IP) algorithms. [1].
A branch of artificial intelligence (AI) called computer vision makes it possible for robots to mimic the human visual system's capacity for precise image analysis and identification. Plant diseases have traditionally been identified and categorized using machine learning (ML) approaches; however, current developments in deep learning (DL) provide substantial promise for improving accuracy in this field of study. Several DL architectures and visualization methods have been used to efficiently identify and categorize the symptoms of plant diseases.

Computer vision-based technologies are being quickly adopted by various industries, including medical diagnostics, espionage, satellite imaging, and farming, demonstrating their advantages. Computer vision systems can be used in agriculture to analyze various traits or symptoms in order to identify and categorize plant diseases. The procedure consists of a clearly defined flow of operations, beginning with image acquisition and moving by means of operations such as segmentation, scaling, filtering, feature extraction, and selection. Finally, ML or DL algorithms are used for detection and classification[2].

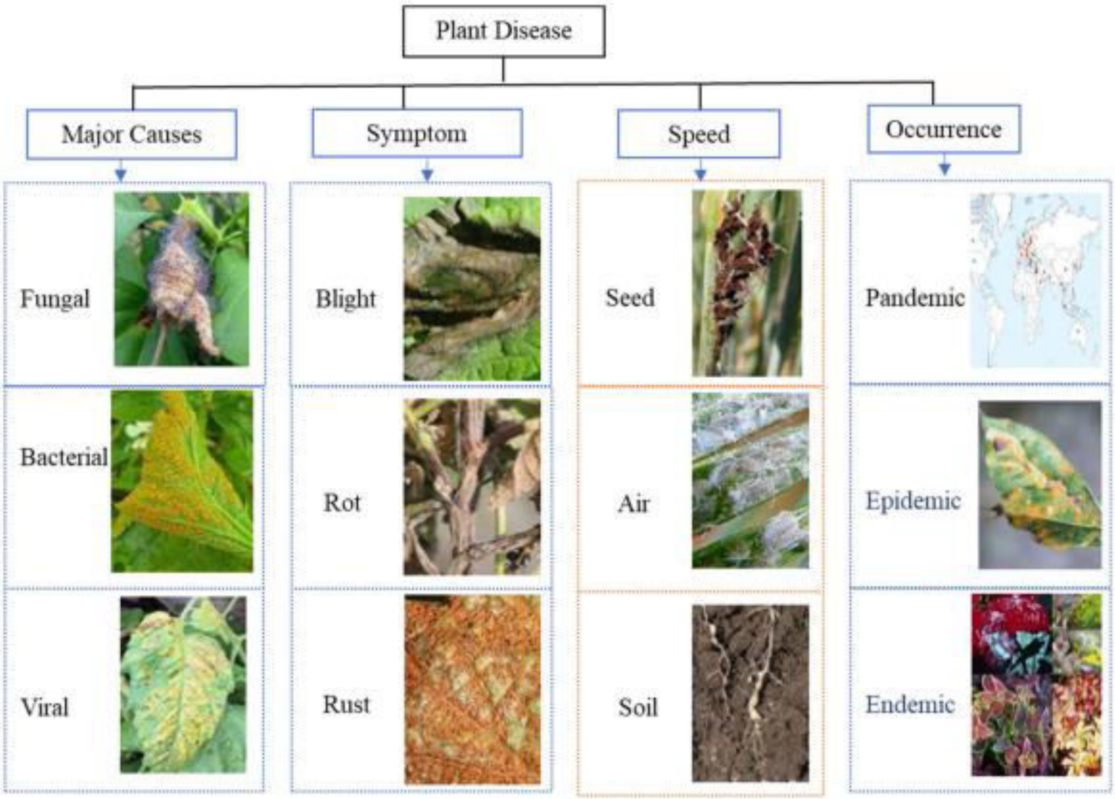


Figure 1: Classification of plant diseases.

II. RELATED WORK

In [3] this study looks at the models and methods used in the literature that currently exists on the identification and categorization of different plant leaf diseases. The work is focused on using deep learning (DL) solutions for real-time insect identification and detection in soybean crops. The study examines how various transfer learning (TL) models perform, including YoloV5, InceptionV3, and CNN, which demonstrate noteworthy accuracy of 98.75%, 97%, and 97%, respectively. Specifically, YoloV5 showed effective real-time detection at 53 frames per second. The authors gathered and labeled a dataset of crop insects from many devices, which lessened the producer's effort.

Utilizing photos from the PlantVillage dataset, the authors of [4] suggest a DL-based approach for categorizing and identifying plant leaf diseases.With 98.29% accuracy in training and 98.029% accuracy in testing across all disease classes, the study used CNN to attain great accuracy.

Based on the size, shape, and color of lesions in leaf pictures, the study in [5] presents a useful technique for detecting and classifying diseases affecting rice plants. By utilizing a fully connected CNN and Otsu's global threshold methodology for picture binarization, the suggested model outperformed previous approaches with an astounding 99.7% accuracy rate on the dataset.

Moreover, [6] offers a CNN-based model that uses a combination of public databases and farm photos to identify and categorize tomato leaf diseases. To avoid overfitting, generative adversarial networks were used, and the outcome was a model that achieved over 99% accuracy in training and test datasets.

Due to differences in picture backgrounds and acquisition settings, the capacity to detect rice leaf disease was limited [7]. This study used two techniques—frozen layers and fine-tuning—to assess how well-known transfer learning (TL) models performed in identifying rice leaf disease. DenseNet169 provided exceptional performance with a testing accuracy of 99.66%, while fine-tuned Xception showed astonishing performance with a testing accuracy of 99.99%.

An innovative DL method for plant leaf disease detection and classification called Ant Colony Optimization with Convolution Neural Network (ACO-CNN) was presented by the authors in [8]. The efficacy of illness diagnostics was evaluated using ACO, and color, texture, and leaf arrangement geometries were extracted from photos using CNN classifier. With respect to accuracy, precision, recall, and F1-score, the suggested ACO-CNN model fared better than the C-GAN, CNN, and SGD models. Compared to other models, the ACO-CNN model outperformed them with an accuracy rate of 99.98%.

Furthermore, for the purpose of detecting plant leaf illness, [9] introduced a DL model (PPLCNet) that included GAP layers, a multi-level attention mechanism, and dilated convolution. The model used augmented meteorological data to increase sample size, which improved feature extraction robustness and generalization. PPLC-Net demonstrated efficiency in accurate and quick recognition with a reduced number of parameters and FLOPs, achieving recognition accuracy and F1-score of 99.702% and 98.442%, respectively.

An efficient CNN model for classifying tomato leaf diseases and identifying disease names affecting tomato leaves was presented in [10]. With a 96% accuracy rate in disease diagnosis, the 2-dimensional Convolutional Neural Network (2D CNN) model with 2-Max Assembling covers and fully connected layers outperformed other classification models such as SVM, VGG16, Inception V3, and Mobile Net CNN model.

III. PROPOSED ALGORITHM

A convolutional neural network (CNN) is implemented for a classification task in the provided code. The Keras framework is used to define the CNN architecture. Here is a summary of the key elements and the employed algorithm:

Architecture Model:
A sequential neural network serves as the model.
Convolutional layers (Conv2D), activation functions (Activation with ReLU), max pooling (MaxPooling2D), batch normalization (BatchNormalization), and dropout (Dropout) are its constituent parts.
The last layers are softmax activation for multiclass classification, fully connected (dense) layers (Dense), and a flattening layer (Flatten).

Losses and Optimizers:
Adam, RMSprop, and SGD (Stochastic Gradient Descent) are the three optimizers that are employed.
MeanSquaredError is taken into account for regression and categorical_crossentropy for classification.

Instruction and Assessment:
The model is trained for 10 epochs with a batch size of 32 using the stated optimizers and losses.
During training, both training and validation data are used.
The metric for evaluation is accuracy.

Preparing data:
Train_test_split is used to divide data into training and testing sets.
The range [0, 1] is used to normalize image data.
To_categorical is used to convert labels into a category representation.

Making a plot:
Accuracy over epochs of training history for various combinations of optimizers and losses is kept track of.
Matplotlib is used to plot the results and display the model's performance.

IV. PSEUDO CODE

Describe the model architecture
Make a model that is sequential.
Include a 32-filter 2D convolutional layer, batch normalization, max pooling, dropout, and ReLU activation.
Convolutional layers should be added in two more sets, with activation, batch normalization, and max pooling coming after each set.
After flattening the output, add a dense layer that includes dropout, batch normalization, ReLU activation, and 512 units.
Include the output layer with softmax activation and three units.

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || Monthly Peer Reviewed & Refereed Journal |

| "1st International Conference on Machine Learning, Optimization and Data Science"|

Organized by

| Department of Computer Science and Engineering, Baderia Global Institute of Engineering and Management, Jabalpur|

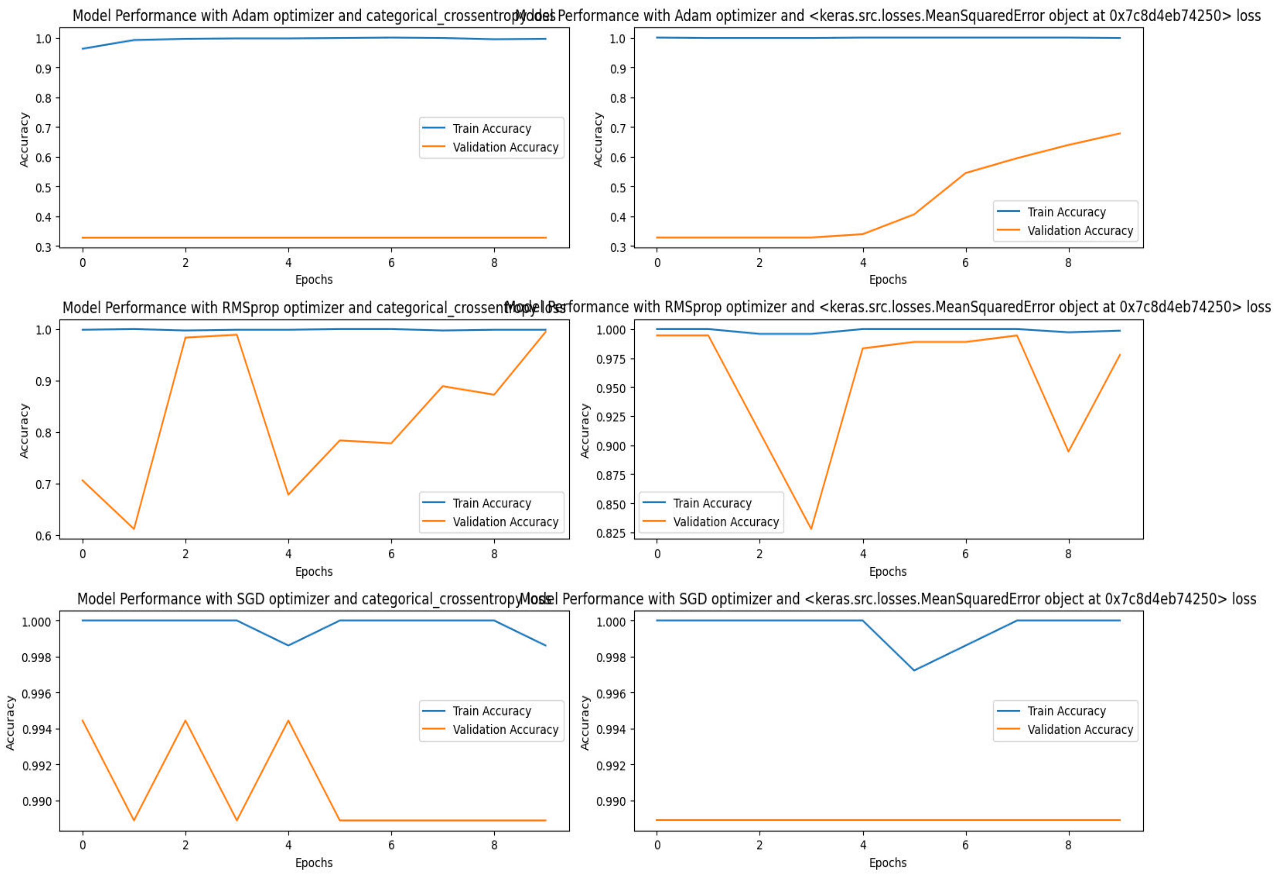
|| Volume 12, Special Issue 1, March 2024 ||

Preparing data:
Divide the dataset into sets for testing and training.
Adjust pixel values to fall between [0, 1].
Resize the data to fit the specified input format.
Apply a category format to the labels.
Losses, Training, and Optimizers:
List the optimizers (Adam, RMSprop, SGD) that have the desired learning rates.
List the names of the optimizers (Adam, RMSprop, SGD).

(categorical_crossentropy, MeanSquaredError) is a list of losses defined.
Go over optimizers and losses again:
Build the model using the optimizer and loss that are in place.
Run the model for 10 epochs, batch size 32, using the training data together with the validation data.
Save the training log for a future plot.
Plot Outcomes:

For every combination of optimizer and loss, create a subplot.
Plot each combination's training and validation accuracy throughout epochs.
For clarity, label the axes and add legends.
Present the plots.

V. SIMULATION RESULTS



International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || Monthly Peer Reviewed & Refereed Journal |

| "1st International Conference on Machine Learning, Optimization and Data Science"|

Organized by

| Department of Computer Science and Engineering, Baderia Global Institute of Engineering and Management, Jabalpur|

|| Volume 12, Special Issue 1, March 2024 ||

VI. Conclusion and Future Work

In this work, we investigated how well various optimization strategies and loss functions trained convolutional neural networks (CNNs) to predict plant diseases. The goal of the study was to improve plant disease prediction models' accuracy by conducting a thorough comparative examination of different optimizer and loss function combinations. We made use of a dataset that included tagged photos of plants in both healthy and unhealthy states.

One of our study's main conclusions was that the convolutional neural network had a test accuracy of 98.89%. This high accuracy shows how well the selected architecture and the optimization methods worked. For agricultural applications, the model's precision in predicting plant illnesses is essential since it aids in disease management and plant health monitoring.

Agricultural Applications' Implications:

The findings add to our knowledge of the best methods for training neural networks in agricultural applications, particularly in the areas of disease prediction and plant health monitoring. Early illness identification can be greatly aided by the application of optimal neural networks, allowing farmers to take prompt preventive action. Through the utilization of cutting-edge technologies like deep learning, machine learning, and computer vision, our methodology is in line with the current trend of applying artificial intelligence to improve agricultural methods.

Prospective Courses:

Prospective investigations may concentrate on broadening the dataset to encompass a more diverse range of plant diseases, hence facilitating a more exhaustive assessment of the model's generalization skills.

The study may also investigate whether the trained model can be applied to other crop species, thereby confirming its relevance in a variety of agricultural contexts.

In conclusion, a possible path forward for improving plant disease prediction in agriculture is the incorporation of artificial intelligence, especially convolutional neural networks. The high test accuracy that was attained supports the sustainable development goals of ending hunger by 2030 and highlights the potential influence of these technologies on global food security.

REFERENCES

1. A_Systematic_Literature_Review_on_Plant_Disease_Detection_Motivations_Classification_Techniques_Datasets_Challenges_and_Future_Trends.pdf.
2. Plant disease detection and classifcation techniques: a comparative study of the performances
3. Tirkey D, Singh KK, Tripathi S. Performance analysis of AI-based solutions for crop disease identifcation detection, and classifcation. Smart Agric Technol. 2023. https://doi.org/10.1016/j.atech.2023.100238.
4. Jasim MA, Al-Tuwaijari JM. Plant leaf diseases detection and classifcation using image processing and deep learning techniques. Int Comput Sci Soft Eng Conf. 2020. https://doi.org/10.1109/CSASE48920.2020.9142097.
5. Upadhyay SK, Kumar A. A novel approach for rice plant diseases classifcation with deep convolutional neural network. Int J Inf Technol. 2022;14(1):185–99. https://doi.org/10.1007/s41870-021-00817-5.
6. Guerrero-Ibañez A, Reyes-Muñoz A. Monitoring tomato leaf disease through convolutional neural networks. Electron. 2023;12(1):1–15. https://doi.org/10.3390/electronics12010229.
7. S. C. Gopi and H. K. Kondaveeti. Transfer learning for rice leaf disease detection. 2023. https://doi.org/10.1109/ICAIS56108.2023.10073711.
8. Algani YMA, Caro OJM, Bravo LMR, Kaur C, Al Ansari MS, Bala BK. Leaf disease identifcation and classifcation using optimized deep learning. Meas Sensors. 2023;25:100643. https://doi.org/10.1016/j.measen.2022.100643.
9. Dai G, Fan J, Tian Z, Wang C. PPLC-Net : neural network-based plant disease identifcation model supported by weather data augmentation and multi-level attention mechanism. J King Saud Univ Comput Inf Sci. 2023;35(5):101555. https://doi.org/10.1016/j.jksuci.2023.101555.
10. P. B. R and A. V V. 2022. Tomato leaf disease detection and classifcation using Cnn. 71 4 2921–2930