Performance Analysis of Different Architectures for Plant Pest Detection System

Tanishkaa Chaturvedi 1, S.Shwetha Iyer2

1,2  Student, Mukesh Patel School of Technology, Management, and Engineering, NMIMS University, Mumbai, India

*Abstract-* The motivation of this project is to provide a solution to the ever-increasing pests in plants and various crops that contribute to millions of plants and crops being left diseased leading to resource wastage every year. This project would help houseplant enthusiasts who have their kitchen gardens at home, as well as farmers to detect pests even at the early onset of any kind of plant disease and result in the optimum usage of their precious resources such as water, fertilizers, as well as the time spent in the fields doing hard work for months and improve overall agricultural productivity. The project uses DCNN, Resnet50, Alexnet, and VGG and compares the best accuracy results for the most favorable outcomes. The dataset used is a refined version of the plant village dataset. Alexnet is the best model with an accuracy of around 97% on the training dataset while the other models have around 90%.

***Keywords—plants, crops, farmers, pests, optimum, DCNN, Resnet50, Alexnet, VGG,***

1. INTRODUCTION

A substantial portion of India’s agricultural land is suffering from soil degradation, including soil erosion, nutrient depletion and loss of organic matter. The root cause of this issue lies in the imbalanced fertilizer use, and other intensive farming practices to maximize profits. The farming industry is no longer food-producing but merely a chemical industry. In this condition, if we can detect pests from the early onset, then a lot of unnecessary chemical usage can be prevented. By implementing weekly drone checks with the camera having the plant pest detection algorithms, a lot of problems in the field can be prevented.

Plant pest detection is the recognition of an external entity harming or slowing down the growth of plants. The usual way of dealing with this is farmers infesting and spraying the crops with harmful chemicals from the beginning, leading to the plants losing their natural nutrients, potency and purity. This is one big reason contributing to autoimmune diseases spreading in humans in recent times more than before and messing up with the natural potency and immunity of humans. The aim is primarily to consistently check the plants for early onset of any kind of disease, so they can be saved midway or before, using precise medicines/fertilizers and help prevent from being completely lost to the disease.

Apart from this, the aim is also to reduce fertilizer usage in plant fields so that it is not chemical farming but more of an actual food farming. A lot of time is spent using multiple types of fertilizers and using trial and error methods to find out which specific fertilizer or chemical works for the specific disease. In this, the soil is also harmed in the process. By the early detection of pests, and making sure of what the accurate disease is from the beginning, unnecessary guesswork and excess usage of chemicals in the soil can be prevented.

Agriculture is a major pillar when it comes to maintaining life on earth. A significant amount of work has been accomplished and is constantly being advanced in the field of plant pest detection. The aim is to detect and manage plant pests more effectively to minimize crop damage and losses. For instance, [1]remote sensing and satellite imagery are being used to monitor large agricultural areas. These images can help identify changes in crop health and detect patterns associated with pest infestations.[2] Internet of Things (IoT) technology and sensor networks are used to collect data on environmental conditions and plant health. These sensors can detect early signs of stress or pest activity in crops. [3]DNA sequencing technologies are used for identifying specific pests and diseases by analyzing the genetic material of the pathogens. This approach is especially valuable for diagnosing diseases caused by bacteria, fungi, and viruses. Lastly, [4]automated phenotyping platforms combine robotics, computer vision, and machine learning to assess plant health and identify pest damage, nutrient deficiencies, and other stress factors.

The challenge that we have noticed in the work conducted in this field so far is not in the detection of pests, but in the ease of access and availability of such technologies to each Indian farmer in the remotest of areas where farming is widely practiced but something even as basic as electricity is scarce. So our goal would not only be to make a successful plant pest detection technology but to also make it user-friendly and widely available for the vast Indian population.

We've delved into a fascinating research project that revolves around classifying various algorithms. Our main goal is to use a consistent dataset featuring 27 different categories of plant images, encompassing both healthy and unhealthy specimens. For our project, we've chosen to employ four distinct algorithms: DCNN (Deep Convolutional Neural Network), ResNet50, VGG16 (Visual Geometry Group), and AlexNet models.

We intend to conduct a comparative analysis of these algorithms, all while utilizing the same plant types. This study will provide us with valuable insights into the precision and practicality of each algorithm. Ultimately, this research will guide us in identifying the most appropriate algorithm for specific plant classifications.

Our proposed model compares 4 different algorithms in depth and the results achieved through them all. The future scope is to build a real-world tool that can be widely available in the remotest areas of India and help prevent plant loss and save soil’s natural potency in the long run.

1. LITERATURE REVIEW

Anwar Abdullah et al. (2002) [5] implemented a convolutional neural network VGG 16 model for plant pest detection for which 19 classes of plant diseases were chosen from the Plant Village dataset. A total of 15,915 leaf images were taken and an accuracy of 95.2% with the testing loss being 0.4418.

Beibei Wang et al. (2023) [6] proposed an ultra- lightweight efficient network (ULEN) that consists of two parts, a deep feature extraction module that adopts residual depth-wise convolution and a classification module receiving multi-scale features enhanced by a spatial pyramid pooling layer. The network is constructed in a very compact design with approximately only 100 000 parameters, and it was tested on two plant datasets to validate different scenarios. The network achieved an accuracy of 98% and performed better than most models.

J. Arun Pandian et al. (2002) [7] have proposed a 14-layered deep convolutional neural network that was trained inmulti-graphics processing units (MGPUs) environment for 1000 epochs. The authors have used image augmentation techniques like BIM, DCGAN, and NST to enhance the dataset which comprises of 147,500 images of 58 different healthy and diseased plant leaf classes and one no-leaf class. On the test images, the model achieved 99.9655% overall classification accuracy, 99.7999% weighted average precision, 99.7966% weighted average recall, and 99.7968% weighted average F1 score. The authors note that the model performs much better as compared to existing transfer learning approaches.

Natheer Khasawneh et al. (2022) [8] have considered tomato leaf images from the Plant Village dataset and have implemented these models- DarkNet-53, DenseNet-201 , GoogLeNet, Inceptionv3, MobileNetv2, ResNet-18, ResNet-50, ResNet-101, ShuffleNet, SqueezeNet , and Xception using transfer learning. The experiments were performed 10 times to account for randomness and the ten categories were mean values of 99.3% precision, 99.2% F1 score, 99.1% recall, and 99.4% accuracy.

Qingcong Lv (2023) [9] implemented and trained five pre-trained deep learning models and fine tuned them using transfer learning on different grape leaf varieties at maturity. The author used two voting ensemble ML models to combine the predictions from the five models and the ensemble classifier gives the highest accuracy of 98.1 %.

Radhika Chapaneri et al. (2020) [10] performed a comparative study of different deep learning models for plant pest detection. The study can be used to identify the appropriate metrics to be incorporated with the average accuracy range spanning from 87% to 96%. The authors note that the best case scenario was obtained using the Full Colour and Grey Scale model using F1 score achieving an accuracy of 99.84%. The paper notes that the performance metrics were observed in a controlled environment and it does not indicate the performance of the model in real time.

Shengyi Zhao et al*. (2022) [11] proposed an improved deep convolutional neural network , which includes* parallel attention mechanism modules and residual blocks for crop pest recognition in a real agricultural environment.. The model achieves up to 98.17% accuracy and this can be applied in other public datasets.

Shaik Sameer et al. (2021) [12] implemented an enhanced AlexNet model on the plant village dataset. In the modified model, the authors used 3 convolutional layers with kernel size 3x3 and the last three layers were replaced by a fully connected layer, a softmax layer , and a classification output layer. The paper suggests a very good performance of the model on the test data as compared to existing models.

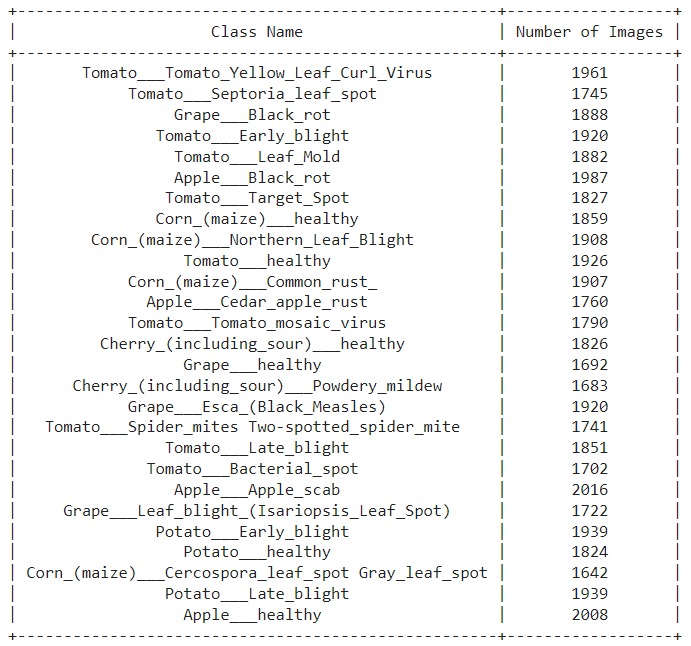
Umamaheswari S and Pragadesh N R (2023) [13] implemented two algorithms - ResNet 50 with custom convolution and identity layers and a modified CNN for plant pest detection using the Plant Village dataset. The study suggests that the ResNet 50 is a better model with an accuracy of 96.99% than modified CNN with an accuracy of 94.86%.

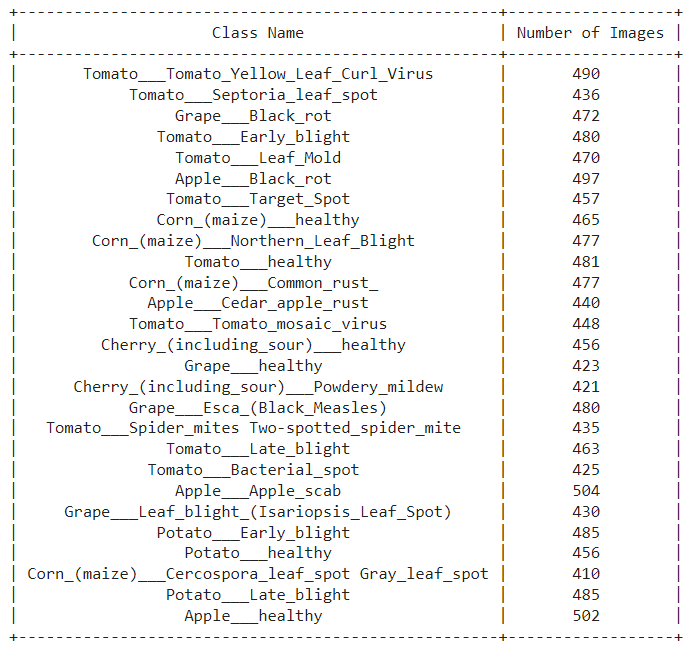
1. DATASET

The dataset that we have used for this project is a refined version of the Plant Village dataset with 38 classes of already split dataset into training and validation, and 33 images in the training dataset from 8 classes. There are 14 leaf plants and while excluding healthy leaves, we have 26 types of images that show a particular disease in a particular plant.

The plant varieties chosen are apple, tomato, potato, corn, cherry and grapes.

Figures 1, 2 , and 3 contain the classes of the following plant images with their quantity for train, valid and test respectively.

   
Figure 1: Training set

  
Figure 2: Validation set

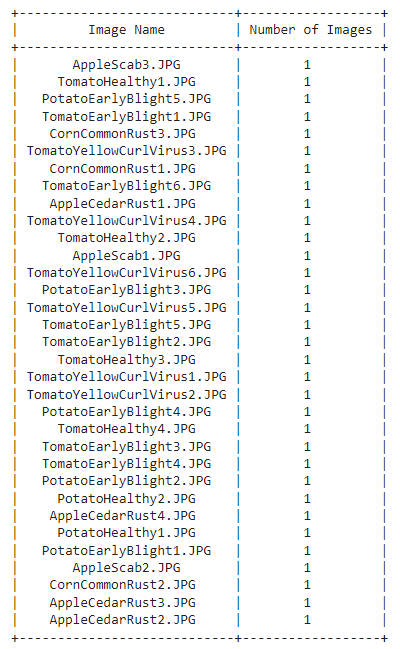


Figure 3: Test set

1. *Data preprocessing*

The code initiates by importing a dataset from a JSON file sourced from Kaggle [14] .11 classes were removed using libraries glob and shutil and the total number of classes is 27.

To enhance the model's performance, image data generators are employed. The images are uniformly resized to a standard 224x224 pixel resolution and the pixel values are normalized. Data augmentation is performed for the training and validation data using the ‘ImageDataGenerator’.

Data augmentation is a technique commonly used to increase the diversity of the training data by applying various transformations to improve the model's generalization ability and robustness. This step is also helpful for those who have limited training data as it greatly increases the dataset size by making variations in the original one. This in turn prevents overfitting and improves the model’s ability to generalize to new data.   
  
We used the generators to rescale the images and apply shear, zoom, and other data augmentation techniques enhancing the model's ability to learn from diverse data.  
  
Figure 4 shows the distribution of images for each of the 27 classes

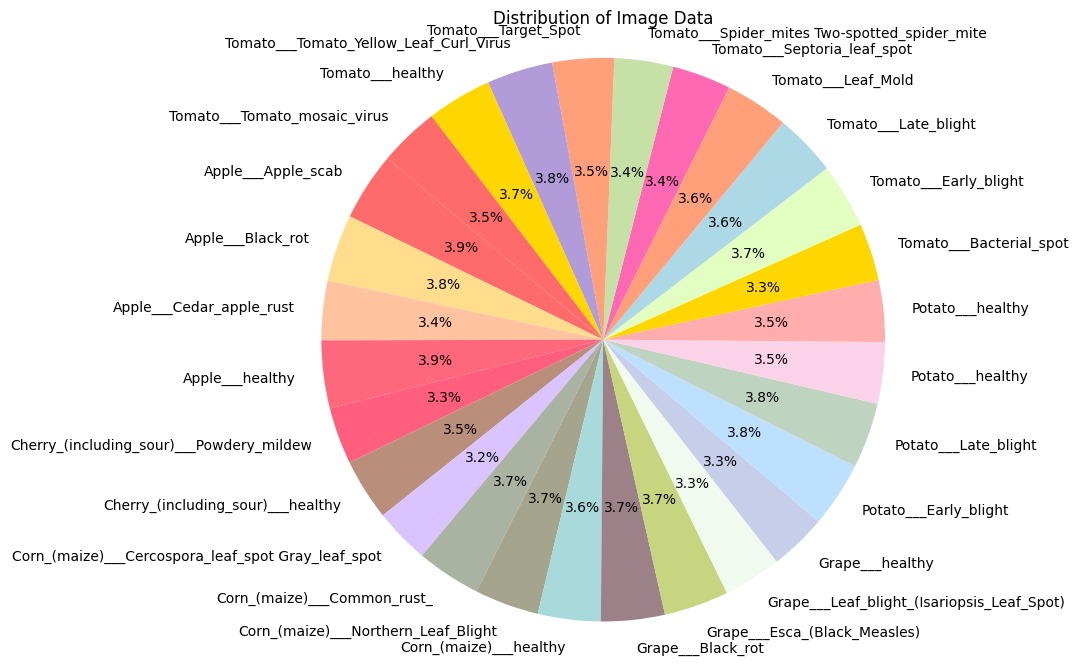


Figure 4. Pie chart of the dataset showing data is balanced

1. PROPOSED MODEL

The workflow comprises of the following structures:

1. Data Pre-processing and dataset preparation: The input images of plants are first pre-processed to enhance their quality and remove any noise or irrelevant information.
2. Model Selection and Training: The models taken into consideration are deep convolutional neural network (DCNN) , ResNet50, AlexNet, and VGG16, which are considered for the classification task. These models are trained on the prepared dataset, allowing them to learn the distinctive features that differentiate healthy plants from infected ones.
3. Classification and Treatment: Once trained, the selected DCNN model is used to classify new plant images. The model analyzes the features of the input images and predicts whether the plant is healthy or infected. Based on the classification results, appropriate treatment measures can be implemented for infected plants.

The proposed model aims to provide an automated and efficient method for plant disease detection and classification, potentially aiding in crop management and early disease intervention.

Figure 5 shows the overall workflow of the paper

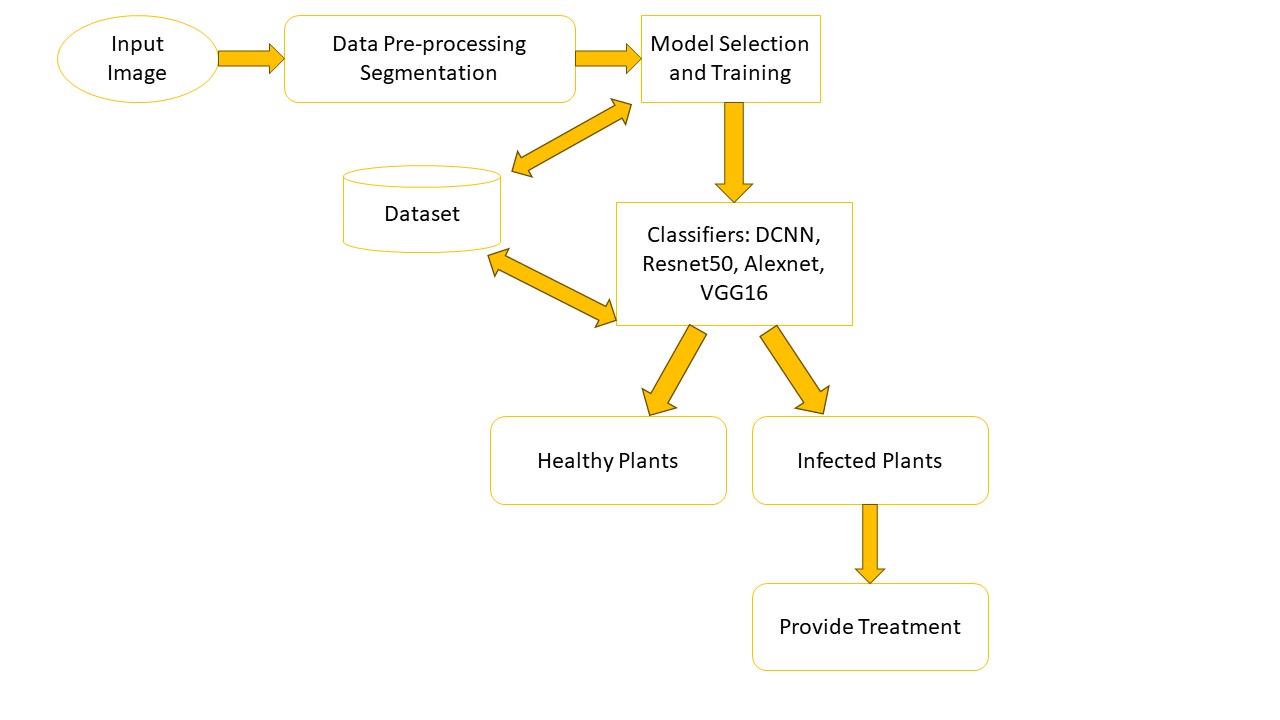


Figure 5: Workflow of the paper

1. *VGG16 model*

The model uses pre-trained weights of the VGG16 model on CNN using transfer learning. The top fully connected layers of the original model that are used for ImageNet classification are excluded since we need to add our custom classifier on top. The pre-trained weights are set to be non-trainable, meaning that the weights in these layers cannot be updated during training, allowing us to use pre-trained features without any changes. A new sequential model is created and the pre-trained weights are added to the custom model. A flatten layer is added to convert the 2D feature maps into a 1D vector. A dense layer with 27 units and a softmax activation function is added for multi-class classification. Early stopping is defined as a callback as it monitors the validation loss and stops training if the loss doesn't improve for 5 epochs. It's also responsible for restoring the best model weights based on the lowest validation loss.

Figure 6 displays the architecture of the VGG-16 model [14].

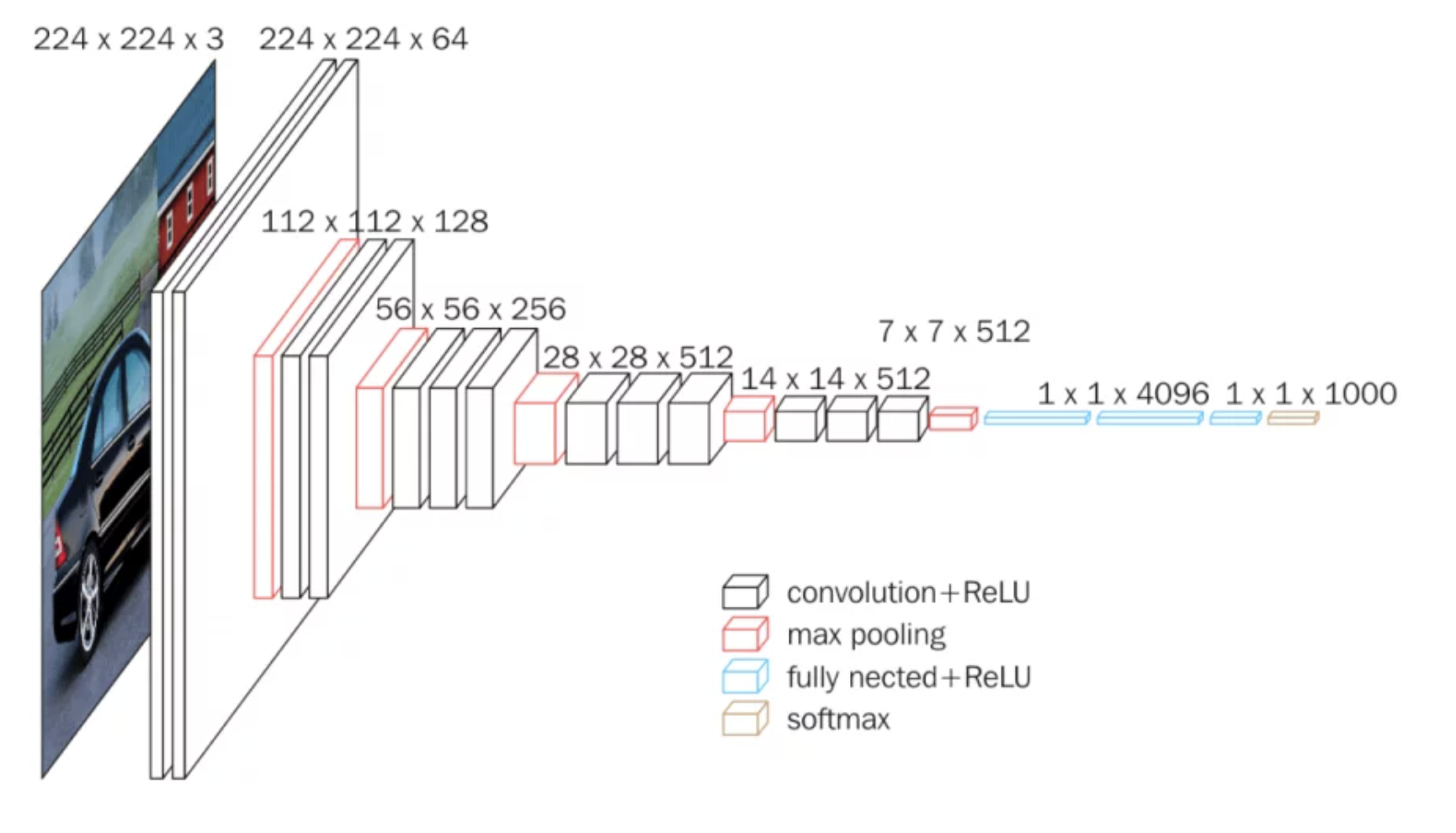


Figure 6: Architecture of VGG16

1. *AlexNet model*

Several convolutional layers are added to the model. Each ‘Convolutional2D’ represents a convolutional layer added as a filter to the input. For example, Convolution2D(96, 11, strides=(4, 4), padding='valid', input\_shape=(224, 224, 3), activation='relu') applies 96 filters of size 11x11 with a stride of (4, 4) and 'valid' padding to the input image. The ReLU activation function is used. Batch normalization is added after each convolutional layer. Batch normalization normalizes the activations of the layer to make training more stable and improve the convergence of the network. ‘MaxPooling2D layers perform max-pooling on the input feature maps. Max-pooling reduces the spatial dimensions and helps capture the most important features. Batch normalization is applied after each max-pooling layer. The flatten layer flattens the 2D feature maps into a 1D vector. This is a necessary step to connect the convolutional layers to the fully connected layers. Each Dense layer specifies the number of units and the activation function. Dropout layers (Dropout(0.4)) are added with a dropout rate of 0.4 after some of the fully connected layers. Dropout is used to prevent overfitting.

Figure 7 displays the architecture of the AlexNet model [15].

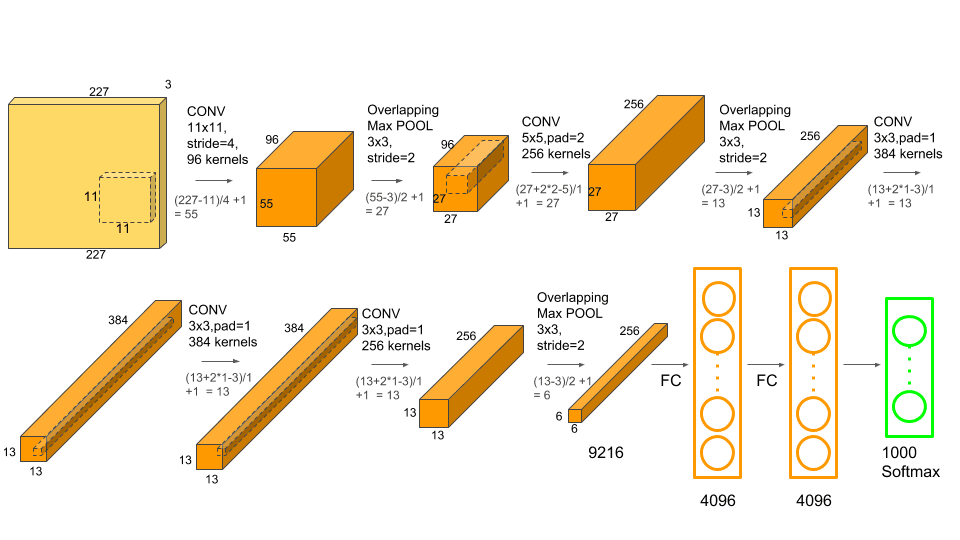


Figure 7: Architecture of AlexNet

1. *DCNN model*

We have implemented the model as per the paper ‘Plant Disease Detection Using Deep Convolutional Neural Network’ [13].Five convolutional and five max-pooling layers were used to develop the proposed 14-DCNN model.

Figure 8 shows the architecture employed in the paper titled ‘Plant Disease Detection Using Deep Convolutional Neural Network’ [7].

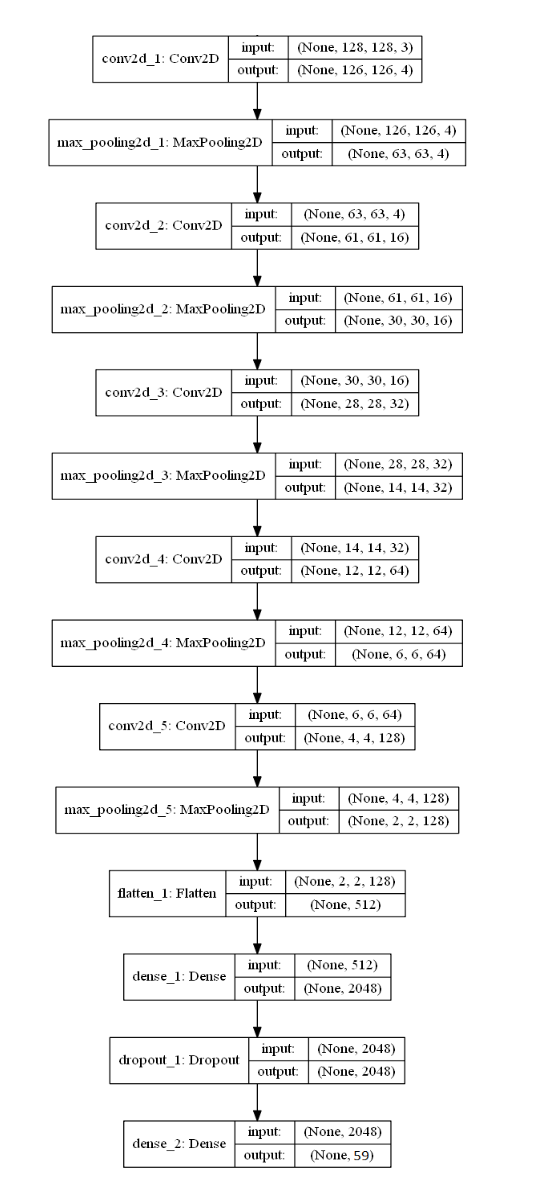
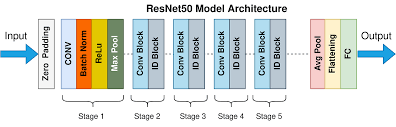


Figure 8: DCNN architecture

1. *Resnet 50 model*

The model deployed consists of several layers, each performing a specific task in the classification process. The first layer, base\_model\_tf(x, training=False), loads the pre-trained ResNet50 weights and extracts features from the input image. The GlobalAveragePooling2D() layer reduces the spatial dimensions of the output feature map to a single vector, making it easier for subsequent layers to process. The next two layers, Dense(128, activation='relu') and Dense(64, activation='relu'), are fully connected layers that transform the feature vector into a more suitable representation for the final classification task. The ReLU activation function introduces non-linearity into the model, allowing it to learn complex decision boundaries. Finally, the Dense(27, activation='softmax') layer outputs a 27-dimensional vector representing the probabilities for each of the 27 categories. The softmax activation ensures that these probabilities sum up to 1.Figure 9 displays the architecture of the Resnet 50 model [16]

  
 Figure 9: Resnet 50 architecture

1. RESULTS AND DISCUSSIONS

Accuracy, precision, recall, F1-score, TPR, and FPR are the parameters taken into consideration to evaluate the models.

Table 1 shows the accuracy of each model deployed on the training data.

|  |  |
| --- | --- |
| **Model** | **Accuracy** |
| VGG16 | 91 % |
| AlexNet | 97% |
| DCNN | 67% |
| Resnet 50 | 94% |

Table 1: Performance measure of each model using the accuracy metric.

Alexnet is the best model with an accuracy of around 97% on the training dataset while ResNet shows an accuracy of 94%, VGG16 with 91% and DCNN with 67%. More epochs have to be run to determine the actual performance of the models since the conditions were controlled and the results may vary in real-time environment.

VI. CONCLUSION AND FUTURE SCOPE

We trained 4 different models namely DCNN, Resnet 50, Alexnet and VGG16. The accuracies obtained have been specified in Table 1. The Resnet model was trained for 10 epochs, DCNN for 5, while Alexnet and VGG16 for 3 epochs using Adam optimizer, due to computational resource restrictions. The input images were resized to 224x224 pixels and pre-processed using data augmentation and rescaling. We used a training set of 49865 images belonging to 27 classes and a validation set of 12465 images belonging to 27 classes.

Overall our results help us to classify 4 different algorithms demonstrating the effective of each as well as the importance of data augmentation, dropout and early stopping in improving all the model’s performance as a whole.

Future research aims to examine the proposed system on a larger dataset, compare it with other models, and improve accuracy. It also focuses on image enhancement methods and collecting images of diseases and pests. The main objective of our subsequent research is to develop a website for plant disease detection and visual inspection.

REFERENCES

[1]“Agricultural Remote Sensing Basics — Publications.” https://www.ag.ndsu.edu/publications/crops/agricultural-remote-sensing-basics

[2]“IoT Monitoring System for Early Detection of Agricultural Pests and Diseases,” *IEEE Conference Publication | IEEE Xplore*, Mar. 01, 2018. https://ieeexplore.ieee.org/document/8788860

[3]“Agricultural Biotechnology Glossary,” *USDA*. https://www.usda.gov/topics/biotechnology/biotechnology-glossary

[4]A. Atefi, Y. Ge, S. K. Pitla, and J. C. Schnable, “Robotic Technologies for High-Throughput Plant Phenotyping: Contemporary Reviews and Future Perspectives,” *Frontiers in Plant Science*, Jun. 25, 2021. https://doi.org/10.3389/fpls.2021.611940

[5] A. A. Alatawi, S. M. Alomani, N. I. Alhawiti, and M. Ayaz, “Plant Disease Detection using AI based VGG-16 Model,” *International Journal of Advanced Computer Science and Applications*, Jan. 01, 2022. https://doi.org/10.14569/ijacsa.2022.0130484

[6] B. Wang, C. Zhang, Y. Li, C. Cao, D. Huang, and Y. Gong, “An ultra-lightweight efficient network for image-based plant disease and pest infection detection,” *Precision Agriculture*, Apr. 22, 2023. https://doi.org/10.1007/s11119-023-10020-0

[7] J. A. Pandian, D. Kumar, O. Geman, M. Hnatiuc, M. Arif, and K. Kanchanadevi, “Plant Disease Detection Using Deep Convolutional Neural Network,” *Applied sciences*, Jul. 10, 2022. https://doi.org/10.3390/app12146982

[8] N. Khasawneh, E. Faouri, and M. Fraiwan, “Automatic Detection of Tomato Diseases Using Deep Transfer Learning,” *Applied sciences*, Aug. 24, 2022. https://doi.org/10.3390/app12178467

[9] “Classification of Grapevine Leaf Images with Deep Learning Ensemble Models,” *IEEE Conference Publication | IEEE Xplore*, May 12, 2023. https://ieeexplore.ieee.org/document/10165757/

[10] “Plant Disease Detection: A Comprehensive Survey,” *IEEE Conference Publication | IEEE Xplore*, Apr. 01, 2020. https://ieeexplore.ieee.org/document/9137779

[11] S. Zhao, J. Liu, Z. Bai, C. Hu, and Y. Jin, “Crop Pest Recognition in Real Agricultural Environment Using Convolutional Neural Networks by a Parallel Attention Mechanism,” *Frontiers in Plant Science*, Feb. 21, 2022. https://doi.org/10.3389/fpls.2022.839572

[12] “Pest and Disease Detection from Plant Leaves using Enhanced AlexNet Model,” *IEEE Conference Publication | IEEE Xplore*, Jul. 09, 2021. https://ieeexplore.ieee.org/document/9622569

[13] “Performance Analysis of ResNet50 Architecture based Pest Detection System,” *IEEE Conference Publication | IEEE Xplore*, Mar. 17, 2023. https://ieeexplore.ieee.org/document/10112802

[14] M. U. Hassan, “VGG16 &#8211; Convolutional Network for Classification and Detection,” *Neurohive / Neural networks*, May 09, 2023. https://neurohive.io/en/popular-networks/vgg16/

[15] A. Pujara, “Concept of AlexNet:- Convolutional Neural Network - Analytics Vidhya - Medium,” *Medium*, Dec. 16, 2021. https://medium.com/analytics-vidhya/concept-of-alexnet-convolutional-neural-network-6e73b4f9ee30

[16] S. Mukherjee, “The Annotated ResNet-50 - Towards Data Science,” *Medium*, Aug. 18, 2022. https://towardsdatascience.com/the-annotated-resnet-50-a6c536034758