ISSN: 0976-9102 (ONLINE)
DOI: 10.21917/ijivp.2022.0405

REAL TIME CORN LEAF DISEASE DETECTION USING CONVOLUTION NEURAL NETWORK

Kamesh Sonti, M. Thamarai and P. Sudheer Chakravarthi
Department of Electronics and Communication Engineering, Sri Vasavi Engineering College, India

Abstract

Agriculture is the primary resource of livelihood, and the economy of our country highly depends on agricultural productivity. For this reason, plant disease detection places a vital role in the agriculture sector. According to one survey, in India, nearly 70% of the population depends on agriculture which is composed of many crops. Disease identification in plants is very challenging for farmers as well as for researchers. We proposed a 24-layer deep learning model in our paper using convolution neural networks (CNN) for the detection of corn leaf diseases by using real time image dataset as input. The CNN model is trained with different corn leaf image samples and model performance is tested and is reported with the evaluation metrics. The obtained results are compared with CNN pre-defined models which shows the superior performance of the proposed model compared to other state-of-the-art approaches.

Keywords:

Agriculture, Plant disease detection, Deep learning, Convolution neural networks, Corn leaf image

1. INTRODUCTION

Agriculture sector plays a dominant role in the economic system of a country and is the backbone of industries that involve the processing, promotion, and distribution of agricultural products. The economy of these industries mainly depends on Agriculture only [1]. Agriculture includes forestry, dairy, fruit cultivation, poultry, etc., also horticulture, sericulture, forestry, poultry, fishing, and logging are generating 17% of the country's GDP and providing maximum employment. Nowadays agricultural research and development are mainly focused on increasing the yielding capability of crops [2]. About 210 million acres of land in India are dedicated to agriculture. The popular crops include rice, wheat, corn, jowar, sunflower, etc. and popular fruits include mango, grapes, banana, grapes, apple, pomegranate, guava, etc. After rice and wheat, maize is the third most important crop in India. It is a popular food grain and if it is affected by any leaf diseases, there will be a loss in production occurs which leads to decay of country economy and risk the availability of food grains.

The main reason for decline in the yield of the crop is the occurrence of pathological issues and is a key challenge to control the effect [3]. In the early stages, the diseases in corn leaves manifest on various parts of the plants and exhibit the symptoms of color change, blight, and spots. Thus, the only method to ensure a higher yield is to discover the disease early and prevent it. As a result, the recent technologies in artificial intelligence using digital imaging techniques are used to increase yield and therefore contribute to the growth of the economy of a nation.

Agricultural image processing is one of the core applications and the most growing research area now. Image processing techniques are used in various fields, including agriculture. In agriculture sector, the images are captured by images through cameras, aircraft, or satellites. These images are then processed and analyzed using computers via image processing techniques. It is made easy with new technological advancements in image capture and data processing to solve various problems in the fields of agriculture. The images are used to provide a visual view of pathogens in agriculture and related fields. The study and explanation of plant illness, its symptoms, and visual metrics, as well as the collected photographs of plant components, are useful for observation and analysis [4].

Fig.1. Traditional vs Modern Farming

The farming methods are changed from ancient days to modern days by adopting sophisticated equipment which is shown in Fig.1. The main aim is to use improved instruments and equipment combined with computer vision techniques to detect diseases in the field automatically. By utilizing image processing and machine learning techniques, diagnosis of plant diseases is done in early stages, and it will be extremely beneficial to the agriculture/horticulture industry [5].

1.1 TYPES OF CORN LEAF DISEASES

Corn leaves are affected by fungi and bacteria which exhibit symptoms for the plant diseases. Fungal and bacteria can over winter in crop residue on the soil surface and spread by wind and water [6].

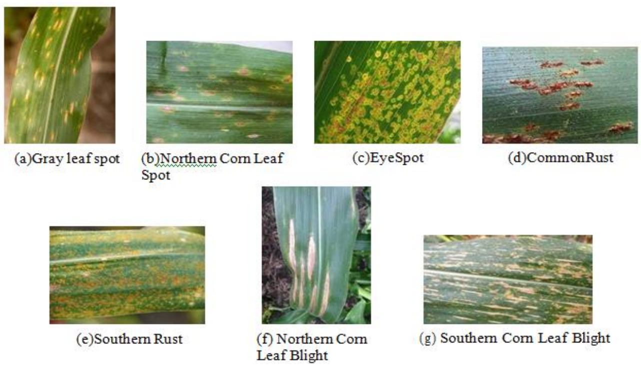


Fig.2. Types of Corn Leaf Diseases

The common types of corn leaf diseases include

- Gray leaf spot
- Eye Spot
- Common Rust
- Southern Rust
- Northern Corn Leaf Spot
- Southern Corn Leaf Blight
- Northern Corn Leaf Blight

The Fig.2 shows the different types of corn leaf diseases that occur regularly. In recent days, many disease identification methods are used in agriculture which are very expensive and can only be utilized by trained persons. These techniques require a lot of work and also need an extreme amount of time and cost. Even though there are automatic detection systems available, it suffers in terms of accuracy which is very much essential in computer-aided detection (CAD) systems [7]. This motivates a low-cost automatic plant disease detection system that identifies the plant disease in a very accurate way.

1.2 AIM AND OBJECTIVE

The main objective of our implementation is to develop an automatic disease detection system to identify corn diseases using deep learning methods. The proposed work involves several phases such as:

- To create a dataset for real-time corn data set consisting of normal and diseased corn leaves.
- To develop a novel CNN architecture for image feature extraction and disease detection.
- The proposed model performance is improved by fine-tuning the model parameters, learning rate, and optimizers.
- To analyze the proposed CNN systems classification performance with various existing pre-trained CNN models.

The proposed work was developed for the classification and recognition of the major diseases in corn leaves such as blight, common rust, and gray leaf spot diseases. because the image data set collected are natural from a cornfield, they are complicated and provide many issues such as uneven illumination, shadowing, lighting, etc., which are difficult to implement. But the proposed CNN method overcomes all the issues and performs classification with maximum accuracy.

The rest of the paper is organized as follows: Section-2 reviews the existing literature on corn leaf disease detection. Section-3 introduced the concept of transfer learning and CNN architecture for image classification. Section-4 describes the proposed methodology and various CNN architectures used for corn leaf disease detection. Section-5 summarizes the simulation results obtained, and a comprehensive analysis of the proposed method. Finally, section 6 describes the conclusion.

2. RELATED WORKS

This section summarizes the existing literature on disease detection in agriculture and different models used in the identification of different diseases in agriculture.

Firouzabadi et al. [8] proposed a supervised learning-based approach for crop disease identification based on the estimation of region and defined an algorithm for effective object area identification. Haiguang Wang et al. [9] proposed a disease identification method with radial basis function (RBF) neural network, generalized regression networks (GRNN) and probabilistic neural networks (PNN) models from the wheat strip under the rust. In [10], another hybrid approach is proposed by the authors using principal component analysis (PCA) and neural networks for plant disease identification by analyzing the features and accurate detection of diseases and prediction is made based on the color as well as other features analysis.

Varshney et al. [11] analyzed different image processing techniques and provides a survey of various techniques for leaf disease detection by analyzing the processing speed and accuracy. Also, they made a comparison of different techniques which gave different results on several datasets. DeChant et al. [12] developed a system for identification of northern leaf blight lesions of corn plants by using a pipeline of CNNs. They performed training on small regions of images and tested the images and evaluated accuracy to access the performance of the CNN.

Ferentinos et al. [13] implemented their approach using various CNN models for plant disease detection. They trained five pre-trained CNN models, and the performance of the models was tested by dividing the image dataset in an 80:20 approach and evaluating the performance metrics. Ni et al. [14] proposed an automatic maize inspection machine by integrating several designs in software and hardware components. They pre-processed the maize kernels with k-means clustering and used a deep CNN model ResNet to train the model and evaluated the testing performance.

A deep learning-based method is proposed in [15] for corn leaf disease detection by tuning the hyperparameters. The model was dumped into raspberry pi-3 and evaluated the accuracy metric to access the performance of the CNN model.

Helong Yu et al. [16] used a combination of k-means clustering and deep learning models to diagnose common leaf diseases. They investigated different pre-defined CNN models by varying different k-values and compared the efficiency of the models by evaluating the diagnostic accuracy values.

Muthusamy et al. [17] implemented a method for sugarcane crop disease detection using diversified deep learning architecture (DDLA) and performance of their model was compared with the pre-trained CNN models with the help of evaluation metrics.

From the literature, we observed that the authors used different datasets, different classifiers, and different features to identify plant diseases, and these are varying time to time. Many of the researchers identified different plant diseases by adopting new technologies in image processing such as CNN models. They used either pre-trained CNN models or developed a new CNN architecture and trained the models by tuning the hyper parameters and testing the images to evaluate the model efficiency.

Hence, in our paper, we detected corn leaf diseases with the help of CNN models. Various pre-defined CNN models such as AlexNet, VGG, Inception, and ResNet-50are adopted and trained the models, and testing performance is assessed with the help of evaluation metrics. The adopted methodologies are briefly discussed in the next section.

3. CONCEPT OF TRANSFER LEARNING

This section describes the concept of transfer learning and how transfer learning is achieved with CNN models and the architecture of CNN used in image processing applications.

3.1 TRANSFER LEARNING

The concept of transfer learning is to improve the performance of target learners by transferring the related source domain knowledge. It is a machine learning technique where a model trained on one task and is validated on another related task [18]. It can be used in two approaches: i) Develop model approach and ii) pre-trained model approach and is implemented with the deep learning models, especially with CNNs for image processing applications. The concept of transfer learning is shown in Fig.3.

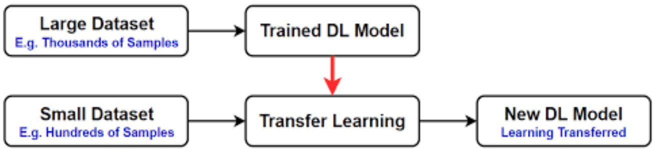


Fig.3. Concept of transfer learning

3.2 TRANSFER LEARNING WITH CNN

CNN is rapidly increasing its impact in transfer learning applications, especially in image processing applications. CNN contains several layers such as convolution layers, normalization unit, and non-linear activation units like rectified linear unit (ReLU), and pooling layers. Various dominant features are extracted from these layers and performed classification using the final layers fully connected (FC) layer and softmax layer [19]. Also, various metrics are used to evaluate the performance of the CNN model. The basic CNN architecture is shown in Fig.4.

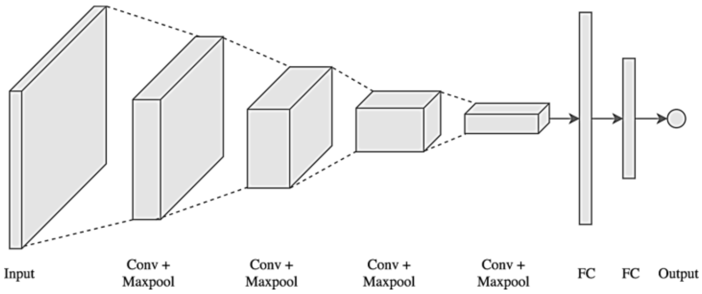


Fig.4. Architecture of CNN Model

3.2.1 Convolution Layer:

The major building blocks in CNN architecture are convolution layers. Convolution is a basic filter operation performed on an image that results in activation. This convolution layer extracts various features from the given input images. Repeated application of the same filter results in activation maps which indicate the strength and position of a detected feature in an image.

Let us consider an image I with dimension $H_k \times W_k \times C_k$ and convolution of the image with a kernel K, results an output $O(u,v)$ defined as:

$$O(u,v) = \sum_{p=-A}^{A}\sum_{q=-A}^{A} I(\mathrm{u}-p_1, \mathrm{v}-\mathrm{q}_1).\mathrm{K}(\mathrm{p},\mathrm{q}) + \mathrm{b}_i \quad (1)$$

where, b_i is the bias term and A represents the size of the convolution kernel. The size of activation map output is $Ho \times Wo \times Co$ and is given by

$$H_0 = \left\lfloor \frac{H_k + 2P - A}{S} + 1 \right\rfloor$$

$$W_0 = \left\lfloor \frac{W_k + 2P - A}{S} + 1 \right\rfloor$$

where, P is the padding size, S represents the stride, and C_k and C_o are of same dimension.

In CNN, the normalization layer and activation function are appended after convolutional layers. The normalization techniques are introduced to make the decision unbiased by an attribute. There are several normalization techniques are available for CNN applications, among those batch normalization technique is implemented in our approach [20]. In batch normalization, the input samples are fed to the training process in the form of batches.

In this technique, all the identical channels from all training examples are in the same batch and grouped. Each channel is associated with mean and standard deviation values. Using these values, the normalization of data $\hat{X}$ is performed as

$$\hat{X} = \frac{X - \mu_c}{\sqrt{\sigma_c^2 + \varepsilon}} \quad (2)$$

where, X represents the feature element in channel c, ε is a constant, and μ_c, σ^2 represents the mini batch mean and variance respectively.

After normalizing the data, the mean and standard deviation values are trainable using scaling and shifting parameters γ and β respectively. These parameters need to be tuned during the gradient descent approach and make the distribution converge faster during the training process.

$$y_j \equiv BN_{\gamma,\beta}(\mathrm{X}_j) = \gamma\,\hat{\mathrm{X}} + \beta \quad (3)$$

where, BN represents batch normalization, $\hat{X}$ is the normalized data.

The normalization layer is followed by a non-linear activation function ReLU [21]. It is an element-wise linear function defined as $f(k) = \max(0,k)$. The advantage of ReLU is it initializes with a positive bias and there is no saturation problem for large input. It makes the gradient computation simple and improves the training speed.

3.2.2 Pooling Layer:

In CNN, dimensionality is reduced by using the pooling operation by replacing the output node at specific locations with a statistic of adjacent locations. The different pooling operations can be max, min, average, sum, etc.

In our approach, we implemented a max-pooling operation on the feature maps. Max pooling report the maximum output value within the given neighborhood. The pooling layers down sample each feature map independently by keeping the depth intact.

3.2.3 Fully Connected Layer:

The activation maps from the final convolutional/pooling layer are connected to one or more FC layers. FC layer combines all the extracted features from the previous layers for larger image pattern identification. The number of output nodes of the FC layer denote the number of output classes.

The output of the FC layer is normalized by a softmax activation layer. It is a mathematical function that maps the neuron output over the output classes with the probability distribution. The activation function is given by

$$L = -\sum_{i=1}^{N}\sum_{j=1}^{K} t_{ij}.\log_2(\mathrm{y}_{ij}) \tag{4}$$

where, N denotes the number of samples, K is the number of classes, t_{ij} represents the i^{th} sample belongs to j^{th} class, and y_{ij} represents the i^{th} sample output for j^{th} class.

4. PROPOSED METHODOLOGY

This section discusses the proposed methodology for the identification of corn leaf diseases using CNN with minimum layers. The result is compared with the various existing CNN models using transfer learning. The detailed architecture of the proposed CNN implemented in our approach is shown in Fig.5.

The proposed methodology comprises Input data acquisition, augmentation, feature extraction, training of the model, and classification. The corn leaf images in the datasets are available with different dimensions. The input images are resized to 128 × 128 × 3 dimensions as per the size of the CNN model input layer. The resized images are employed with data augmentation to enhance the dataset size and avoid overfitting. The applied augmentation techniques are re-scale, width shift, height shift, shear, zoom, and horizontal flip.

The feature extraction, training of the model, and classification steps are performed by the CNN network. The proposed CNN model has five convolution layer and each convolution layer is followed by batch normalization and max-pooling layers. The two fully connected layers are added before the output layer in the model to flatten the output of the convolution layers. Finally, the output layer is added with a softmax activation function for leaf disease classification. The total number of layers in the model is 18 and the number of trainable parameters is 67,156 only. The complete layer details of the proposed CNN architecture are provided in Table.1.

The pre-trained CNN models are taken from Keras library applications with image net weights except for the last layer based on the transfer learning concept. We can extract the necessary features from CNN and performed classification using the FC layer and output layer in the case of pertained models.

In our approach, we are classifying three types of diseases: blight, common rust, and gray leaf spot other than healthy images. Various performance metrics are evaluated to assess the performance of the proposed architecture and the pre-trained models and compared using the parameters obtained from the confusion matrix. The training accuracy and error plots of the proposed and pretrained models were also analyzed. There are several pre-trained CNN architectures used in the classification of corn leaf diseases which are discussed below.

4.1 ALEXNET

AlexNet is the most popular deep learning model consisting of 25 layers with 227× 227 × 3 input image size and includes five convolution layers, three pooling layers, and one dropout layer [22]. The model is developed for the image Net contest and is used to classify 1000 classes of images in the ImageNet database. The model is loaded with its pre-trained layer weights except for the output layer. The output layer is modified and trained according to the last required output disease classes. AlexNet is trained with a batch size of 32, a maximum of 20 epochs, and a 0.00001 learning rate with stochastic gradient descent with a momentum optimizer. The AlexNet architecture is shown in Fig.6.

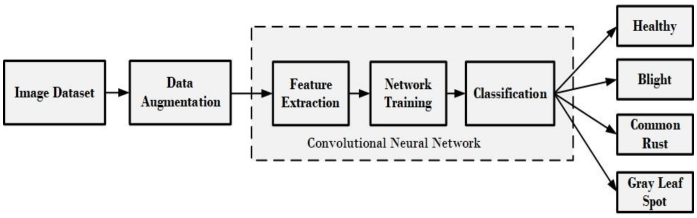


Fig.5. Block Diagram of the Proposed Model

Table.1. Proposed architecture layers

Layer	Number of features maps	Stride	Activation shape	Total learnable parameters
Input	3	-	128×128×3	-
Convolution	48	1	126×126×48	1344
Batch normalization	48	-	126×126×48	96
ReLU	48	-	126×126×48	-
Max. pooling	48	2	63×63×48	-
Convolution	48	1	61×61×48	20784
Batch normalization	48	-	61×61×48	96
ReLU	48	-	61×61×48	-
Max. pooling	48	2	30×30×48	-
Convolution	48	1	28×28×48	20784
Batch normalization	48	-	28×28×48	96
ReLU	48	-	28×28×48	-
Max. pooling	48	2	14×14×48	-
Convolution	48	1	12×12×48	20784
Batch normalization	48	-	12×12×48	96
ReLU	48	-	12×12×48	-
Max. pooling	48	2	6×6×48	-
Convolution	48	1	4×4×48	20784
Batch normalization	48	-	4×4×48	96
ReLU	48	-	4×4×48	-
Fully connected	4	-	1×1×4	3076

Fully connected	4	-	1×1×4	20
Softmax	4	-	1×1×4	-
Classification	4	-	1×1×4	-

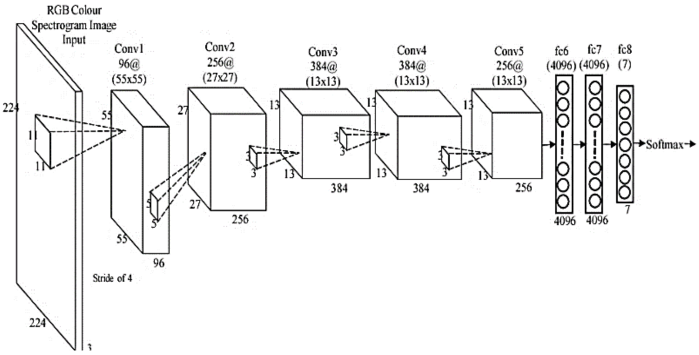


Fig.6. AlexNet Model

4.2 VGG-16

VGG-16 is a CNN having 16 layers with an input size of 224 x 224 RGB image [23]. It is passed through a stack of convolution layers, where the filters were used with a very small receptive field of 3×3 size. The small receptive field is more suitable for the motion of left/right, up/down, and center image details. Spatial pooling is carried out by five max-pooling layers, which follow some of the convolution layers with stride 2. Three FC layers follow a stack of convolution layers in which the first two have 4096 channels each, the third classifies 1000 channels with the softmax layer. The VGG-16 architecture is shown in Fig.7.

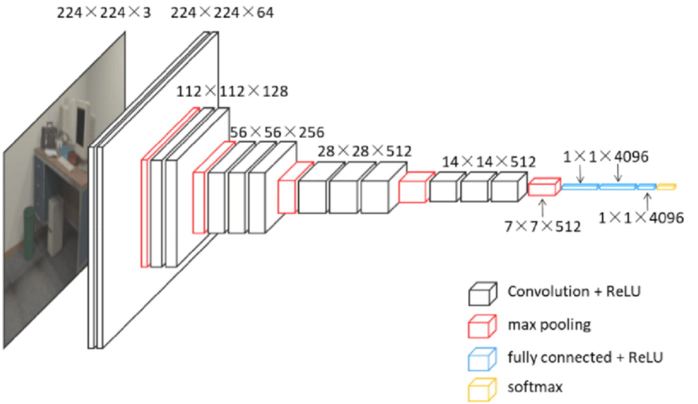


Fig.7. VGG-16 Model

4.3 INCEPTION

GoogleNet (or) Inception is a CNN that is 27 layers deep. It is the core concept of sparsely connected architecture [24]. It uses many filters of multiple sizes that operate on the same level. It involves several stacks of modules called as Inception module. In each Inception module, convolution of size 1×1, 3×3, 5×5, and max pooling of size 3x3 are performed in a parallel way. The layers are stacked together between the input and the output to generate the final output. The Inception network analyzes the objects in multiple scales using these different sizes of convolution filters.

In Inception V2 architecture performs the 5×5 convolution by the two 3×3 convolutions. This reduces the computational time and in turn, increases classification speed. Since the 3×3 convolution is 2.78 less expensive than a 5x5 convolution. Inception V3 has 48 convolutions layers deep network and contains all the features of Inception V2. The network uses 7×7 and batch normalization in Auxiliary classifiers and also prevents overfitting problems with the help of regularizing components and label smoothing. The Inception V3 architecture is shown in Fig.8.

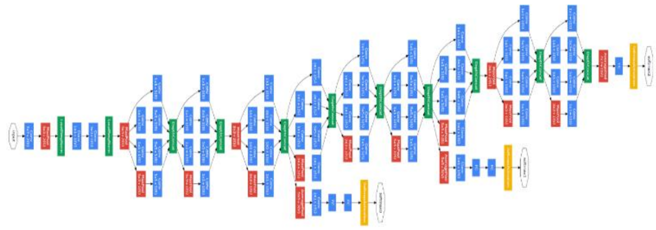


Fig.8. GoogleNet Model

4.4 RESNET-50

ResNet 50 CNN model has 50 layers deep and is most popularly used for image classification. The network has skip/identity connections, which skip one or more layers in the forward pat [25]. The model has 64 different kernels of size 7x7 all with a stride of size 2. The max-pooling with a stride size of 2 is used for feature map reduction. The network ended with a fully connected layer containing 1000 nodes and the softmax activation function is used in the output layer. The ResNet-50 architecture is shown in Fig.9.

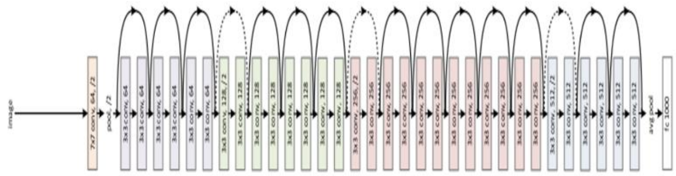


Fig.9. ResNet-50 Model

5. PERFORMANCE METRICS

The performance of the CNN model is assessed by the performance metrics accuracy, sensitivity, specificity, precision and F1-score which are defined as,

$$\text{Accuracy}=\frac{TP+TN}{TP+FP+FN+TN}$$

$$\text{Precision}=\frac{TP}{TP+FP}$$

$$\text{Recall}=\frac{TP}{TP+FN}$$

$$\text{F-score}=\frac{2*TP}{(2*TP)+FP+FN}$$

where, TP=True Positive, TN=True Negative, FP=False Positive and FN=False Negative.

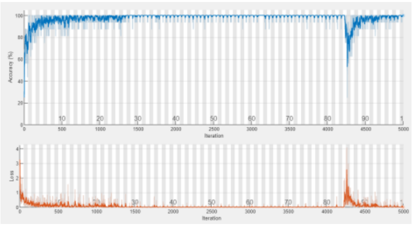
6. RESULTS

This section presents the simulation results carried out by the proposed CNN model and the pre-trained models for the identification of corn leaf diseases. The simulated results mentioned in this section are obtained by executing the proposed model using a system with MATLAB R2021a environment.

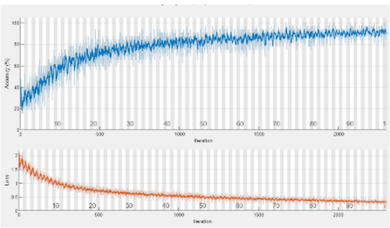
The image dataset contains 807 training images and 261 testing images which include healthy, blight, common rust, and gray leaf spot disease images. These images are augmented to enhance the dataset size and trained the model using various CNN architectures by choosing adam optimizer and varying batch size as 16, 32, 64 and initial learning rates as 0.1,0.01,0.001,0.0001 by limiting the number of epochs as 100. The maximum classification accuracy is obtained with batch size 16, with the initial learning rate of 0.0001 for the given testing dataset. The final softmax layer is modified to classify four classes and tested the images from the test dataset and evaluated the metrics from the confusion matrix.

To understand the advantage of the proposed architecture, the predefined models are also employed for the classification of corn disease images. The obtained results from the proposed model are compared with results obtained from predefined models which are mentioned in Table.2.

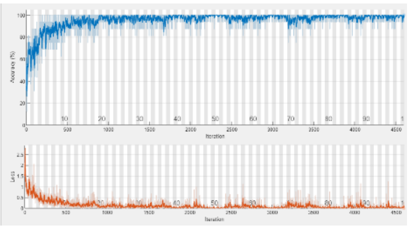
In addition to the above performance metrics, the training accuracy and error plots are also taken and are shown in Fig.10. Among those, Fig.10(a)-Fig.10(d) represent the plots obtained for various pre-trained architectures Alexnet, VGG-16, ResNet-50, and Inception-V3 respectively. These models are trained by training a dataset of corn leaf images by changing the output layer to four classes and testing the model performance with the testing dataset. The Fig.10(e) shows the training accuracy and error plots for the proposed model by tuning the hyper parameters.



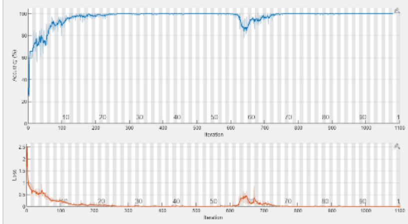
(a) Alexnet



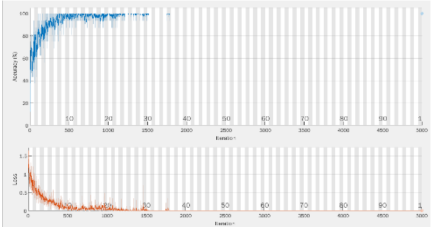
(b) VGG-16



(c) Resnet-50



(d) Inception-v3



(e) Proposed Model

Fig.10. Training accuracy-error plots for various models

Table.2. Performance Metrics Evaluated

CNN Model	Output Class	Accuracy (%)	Precision (%)	Recall (%)	F-score (%)
Alexnet	Blight	88.12	73	77	75
	Common rust	97.32	95	94	95
	Gray leaf spot	88.51	79	76	78
	Healthy	100	100	100	100
VGG-16	Blight	86.21	67	74	70
	Common rust	95.40	92	90	91
	Gray leaf spot	85.06	74	69	72
	Healthy	99.62	98	100	99
ResNet-50	Blight	86.21	67	74	70
	Common rust	95.40	92	90	91
	Gray leaf spot	85.82	76	70	73
	Healthy	99.62	98	100	99
Inception	Blight	85.82	63	74	68
	Common rust	95.4	92	90	91
	Gray leaf spot	85.82	77	70	73
	Healthy	100	100	100	100
Proposed	Blight	91.57	83	83	83
	Common rust	99.23	98	98	98
	Gray leaf spot	92.34	85	85	85
	Healthy	100	100	100	100

7. CONCLUSION

In this paper, we have developed a 24-layer CNN model for the identification of corn leaf diseases from leaf images, and the model performance is compared with the various pre-trained CNN models. Various hyper parameters are tuned to improve the accuracy of the proposed model based on the architecture taken. The model is tested on the test image dataset and results are reported in terms of performance metrics and accuracy and error plots. Hence, we conclude that our proposed CNN model effectively classifies various corn leaf diseases compared to other existing pre-trained CNN models.

REFERENCES

[1] Sneha Adhikari, "Identification of QTL for Banded Leaf and Sheath Blight in Teosinte-Derived Maize Population", *Agricultural Research*, Vol. 65, pp. 1-9, 2021.

[2] Xanthoula Eirini, Dimitrios Moshou and Alexandra A. Tamouridou. "Automated Leaf Disease Detection in Different Crop Species through Image Features Analysis and One Class Classifiers", *Computers and Electronics in Agriculture*, Vol. 156, pp. 96-104, 2019.

[3] P. Ferentinos, Konstantinos, "Deep Learning Models for Plant Disease Detection and Diagnosis", *Computers and Electronics in Agriculture*, Vol. 145, pp. 311-318, 2018.

[4] S.P. Mohanty, David P. Hughes and Marcel Salathe, "Using Deep Learning for Image-Based Plant Disease Detection", *Frontiers in Plant Science*, Vol. 7, pp. 1419-1428, 2016.

[5] Reva Nagi and Sanjaya Shankar Tripathy, "Severity Estimation of Grapevine Diseases from Leaf Images Using Fuzzy Inference System", *Agricultural Research*, Vol. 11, No. 1, pp. 112-122, 2022.

[6] D.R. Smith and D. G. White, "Diseases of Corn", *Corn and Corn Improvement*, Vol. 18, pp. 687-766, 1988.

[7] D.A. Noola and R.B. Dayanand, "Computer Aided Corn Leaf Disease Identification System", *Proceedings of International Conference on Smart Electronics and Communication*, pp. 1-13, 2021.

[8] Firouzabadi, "Performance Evaluation of Supervised Classification of Remotely Sensed Data for Crop Acreage Estimation", *Proceedings of International Conference on International Geoscience and Remote Sensing*, pp. 1-13, 2001.

[9] Haiguang Wang, "Application of Neural Networks to Image Recognition of Plant Diseases", *Proceedings of International Conference on Systems and Informatics*, pp. 1-13, 2012.

[10] Haiguang Wang, "Image Recognition of Plant Diseases based on Principal Component Analysis and Neural Networks", *Proceedings of International Conference on Natural Computation*, pp. 1-14, 2012.

[11] S. Varshney and Tarun Dalal, "Plant Disease Prediction using Image Processing Techniques-A Review", *International Journal of Computer Science and Mobile Computing*, Vol. 5, No. 5, pp. 394-398, 2016.

[12] C. De Chant, "Automated Identification of Northern Leaf Blight-Infected Maize Plants from Field Imagery using Deep Learning", *Phytopathology*, Vol. 107, No. 11, pp. 1426-1432, 2017.

[13] P. Ferentinos, Konstantinos, "Deep Learning Models for Plant Disease Detection and Diagnosis", *Computers and Electronics in Agriculture*, Vol. 145, pp. 311-318, 2018.

[14] Chao Ni, "Automatic Inspection Machine for Maize Kernels based on Deep Convolutional Neural Networks", *Biosystems Engineering*, Vol. 178, pp. 131-144, 2019.

[15] S. Mishra, Rishabh Sachan and Diksha Rajpal, "Deep Convolutional Neural Network based Detection System for Real-Time Corn Plant Disease Recognition", *Procedia Computer Science*, Vol. 167, pp. 2003-2010, 2020.

[16] H. Yu, "Corn Leaf Diseases Diagnosis based on K-Means Clustering and Deep Learning", *IEEE Access*, Vol. 9, pp. 143824-143835, 2021.

[17] T. Muthusamy and Gomathi Eswaran, "Detection of Sugarcane Mosaic Diseases using Deep Learning Architecture to Avoid Annealing Temperature of PCR Primer in Laboratory Testing", *Traitement Du Signal*, Vol. 39, No. 1, pp. 1-13, 2022.

[18] Ian Goodfellow, YoshuaBengio and Aaron Courville, "*Deep Learning*", MIT press, 2016.

[19] Saad Albawi, Tareq Abed Mohammed and Saad Al-Zawi, "Understanding of a Convolutional Neural Network", *Proceedings of International Conference on Engineering and Technology*, pp. 1-8, 2017.

[20] S. Ioffe and Christian Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift", *Proceedings of International Conference on Machine Learning*, pp. 1-13, 2015.

[21] Hidenori Ide and Takio Kurita, "Improvement of Learning for CNN with ReLU Activation by Sparse Regularization", *Proceedings of International Conference on Neural Networks*, 2017.

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet Classification with Deep Convolutional Neural Networks", *Proceedings of International Conference on Advances in Neural Information Processing Systems*, pp. 1097-1105, 2012.

[23] Karen Simonyan and Andrew Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition", *Proceedings of International Conference on Machine Learning*, pp. 1-9, 2014.

[24] C. Szegedy, "Going Deeper with Convolutions", *Proceedings of International Conference on Computer Vision and Pattern Recognition*, pp. 1-8, 2015.

[25] K. He, "Deep Residual Learning for Image Recognition", *Proceedings of International Conference on Computer Vision and Pattern Recognition*, pp. 891-898, 2016.