



# Detection and Classification of Cassava Leaf Diseases using Squeezenet Pretrained Convolutional Neural Network and Support Vector Machine

Arinola I. O., Oke O. A. & Falohun A. S.  
Department of Computer Engineering,  
Ladoke Akintola University of Technology  
Ogbomoso, Nigeria.

E-mails: arinolaio19@gmail.com; aooke@lautech.edu.ng; asfalohun@lautech.edu.ng

## ABSTRACT

Cassava cultivation is important to the nation because of the products and raw materials that are supplied to industries. However, its cultivation is affected by diseases such as cassava brown leaf and cassava mosaic, which have been classified using a Support Vector Machine (SVM). SqueezeNet pretrained Convolutional Neural Network (SPCNN) is adjudged a very effective image detection and classification algorithm with limited usage areas. This research, therefore, was used to detect and classify cassava leaf diseases using SPCNN and SPCNN with multiclass SVM. Performance evaluation of both techniques showed that SPCNN effectively detected and classified cassava leaf diseases with optimal accuracies of 92.22% and 98.89% for SPCNN and SPCNN-SVM, respectively.

**Keywords-** SqueezeNet, SVM, Cassava diseases, Feature extraction, Transfer learning

---

Arinola I. O., Oke O. A. & Falohun A. S. (2026): Detection and Classification of Cassava Leaf Diseases using Squeezenet Pretrained Convolutional Neural Network and Support Vector Machine. Journal of Advances in Mathematical & Computational Science. Vol. 14, No. 1. Pp 1-12 Available online at [www.isteams.net/mathematics-computationaljournal](http://www.isteams.net/mathematics-computationaljournal). dx.doi.org/10.22624/AIMS/MATHS/V14N1P1

---

## 1. INTRODUCTION

Agriculture is one of the backbones of the Nigerian economy with crop production as a means of meeting the food demand of the populace. With increased migration from rural to urban centers, about 70% of the world population was predicted to live in urban cities till 2050 (Sarvary *et al.*, 2012): thus, the need to implement smart agricultural practices to meet increasing food demand (Jiang *et al.*, 2021). One area to implement smart agriculture is in cassava cultivation, however, cassava cultivation is affected by diseases like brown leaf spot, witches' broom, etc. (plantvillage.psu.edu, 2019). These diseases cut down crop yield, reduce the nation's profitability and cause major discouragement to intending cassava farmers and investors. Moreover, the cost and continuous inaccuracy of manual control and management of these diseases are other challenges. Machines are trained to learn, recognize and interpret image features using certain algorithms and statistical models. A few of the techniques in image processing are preprocessing, where images are resized, enhanced and normalized.

Image segmentation is used to partition images for proper focus on the region of interest; and image feature extraction emphasizes the most important character needed to identify the image. Multiclass SVM, K-Nearest Neighbor (KNN) and Convolutional Neural Network (CNN) are some algorithms used for image detection. CNN is a type of neural network that is used in time series and image data. It has three major layers (convolution, pooling and fully connected layers) for operation (Amara *et al.*, 2017) and is efficient for image detection tasks (Kamilaris and Prenafeta-Boldu, 2018). CNN makes smart farming feasible (Gebbers and Adamchuk, 2010) and can be pretrained, trained and retrained for various tasks. Examples of pretrained CNN are SPCNN and AlexNet. SVM is a professional algorithm used for classification tasks. It finds a hyperplane within a data distribution and picks the optimum. SPCNN is an algorithm designed to be small, yet highly accurate. It has fire modules, a layer that combines 1x1 and 3x3 filters to reduce parameters while maintaining accuracy (Anmol *et al.*, 2023). SPCNN with multiclass SVM was used in this research to detect and classify cassava leaf diseases because of its efficient features and the little storage space required. It is capable of distinguishing patterns and making crucial predictions.

## **2. RELATED WORKS**

Numerous studies have been conducted on the classification of crop diseases using a variety of algorithms. Sachin and Patil (2015) carried out detection and classification of disease in plants using Artificial Neural Network and Back Propagation Neural Network (BPNN), results showed that BPNN performed better. Amanda *et al.* (2017) evaluated transfer learning from a deep CNN model for cassava image datasets using Inception v3 and recorded best accuracy of 98.4%. The research showed the use of Inception v3 pretrained CNN on cassava cultivation; however, Inception v3 is not easily deployed and requires larger storage space. Zhang *et al.* (2019) researched fruit classification using image processing and CNN. 13-layer CNN on the dataset was developed. Mingyue *et al.* (2023) implemented a bilinear coordinate attention enhancement module to recognize cotton leaf diseases, achieving 96.61% accuracy. As crop producers seek solutions to product-related challenges, SPCNN, though rarely used, offers high accuracy for image detection while requiring minimal storage space.

## **3. METHODOLOGY**

One thousand two hundred datasets of cassava leaf images were acquired from the Plantvillage database. In addition to these, for validation purposes, twenty (20) live images were captured using Tecno F1 Android with Android version 8.1.0. The dataset was grouped into six (6) classes of cassava leaf diseases and healthy leaves: cassava bacterial blight, cassava brown leaf spot, cassava witches' broom, cassava mosaic disease, cassava streak diseases, cassava brown leaf spot and cassava healthy leaves, with each class having two hundred (200) images. Figure 1 shows the random samples of the dataset. Feature extraction and classification were carried out using SPCNN which is referred to as Transfer Learning (TL). Feature extraction using SPCNN and classification using SVM were also carried out. The flowchart for the two approaches is shown in Figure 2.



Figure 1. Random samples of the dataset

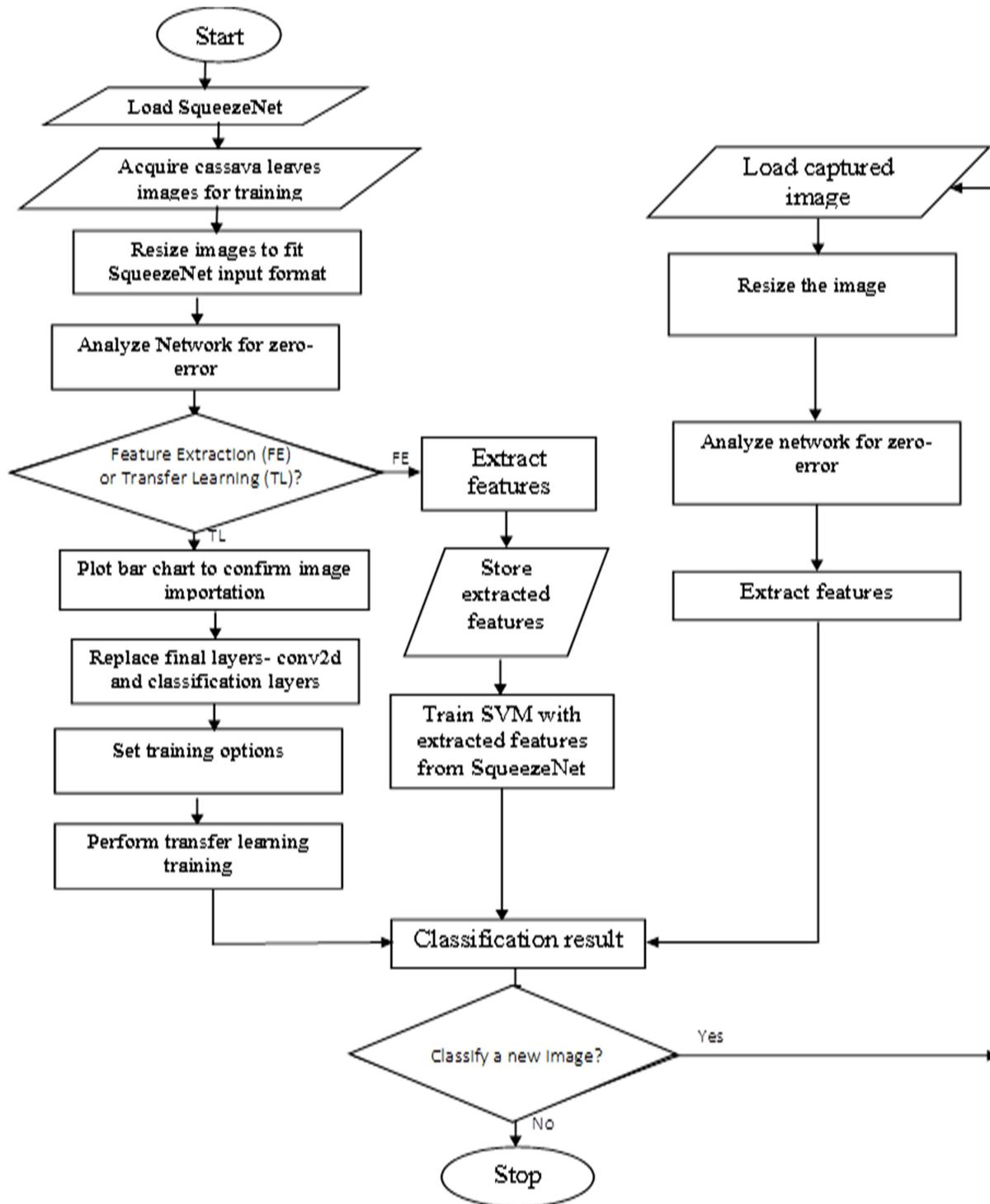


Figure 2. Flowchart of the two learning approaches

### 3.1 Transfer Learning With SPCNN

SPCNN is a pretrained network with an image input size of 227 by 227, which the one thousand two hundred (1200) images dataset conformed to. Also, the partitioning of the dataset into training and testing sets was done to specify the percentage division. Seventy percent (70%) which is 840 images for training and thirty percent (30%) which is 360 images for testing. A bar chart of the imported dataset is shown in Figure 3. The deployment of the SPCNN architecture was facilitated utilizing MATLAB R2020A. The frequency change parameter was utilized for training purposes.

The network's configuration comprises an input layer, followed by a convolutional layer (conv1) that performs kernel convolution on the input image. Subsequently, eight fire modules (fire2-9) are employed, each consisting of CNN building blocks with progressively increasing filter numbers: 16 (fire2-3), 32 (fire4-5), 48 (fire 6-7), and 64 (fire8-9).

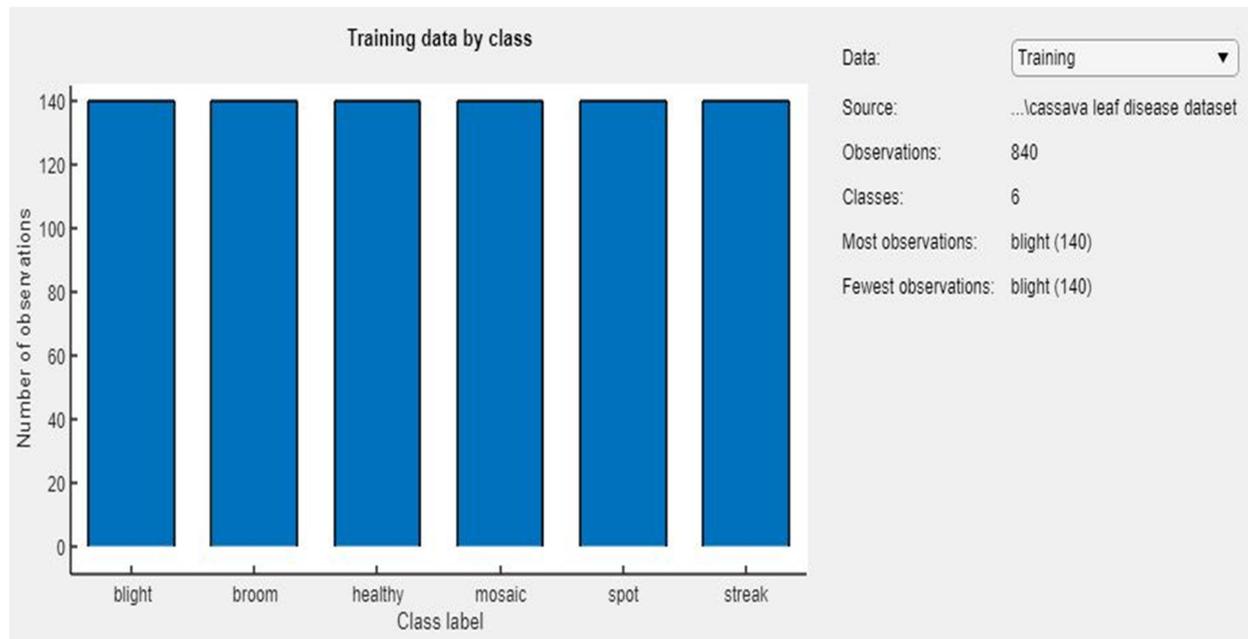


Figure 3. Bar chart of the imported dataset

The SPCNN architecture concludes with a final convolutional layer (conv10) responsible for image classification. The network employs a combination of pooling layers to reduce spatial dimensions: three Max-pooling layers (pool1, pool3, and pool5) with a stride of 2, and a single Average-pooling layer (pool10). These layers down sample the output of preceding convolutional layers, effectively decreasing image dimensionality as shown in Figure 4. Subsequently, the acquired dataset was integrated into the SPCNN model. The network was initialized using the MATLAB command defined in Equation 1.

$$net = SqueezeNet ..... \quad (1)$$

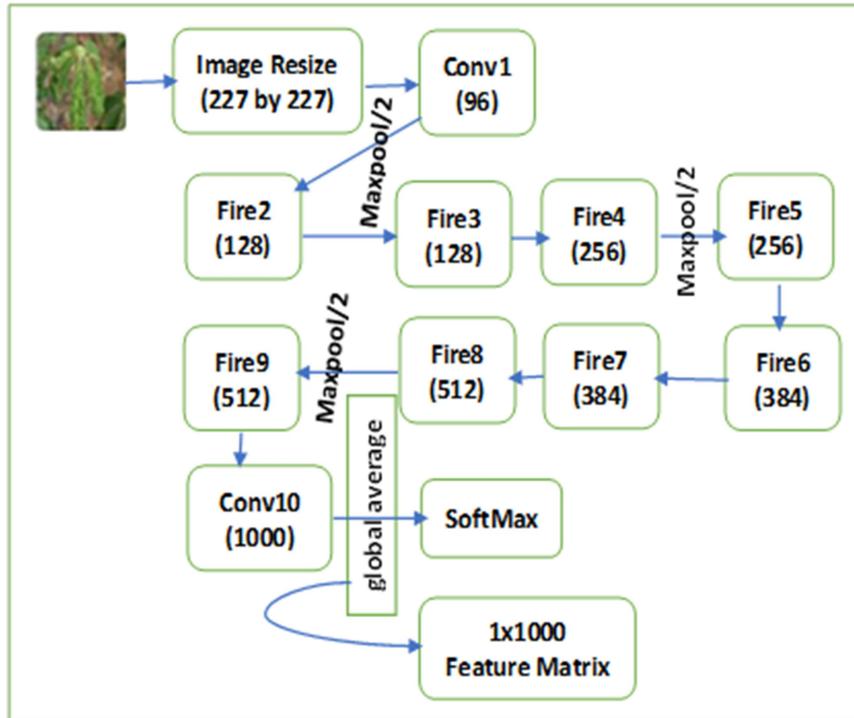


Figure 4. SqueezeNet architecture

The network underwent pretraining on a diverse image dataset. However, to facilitate the new classification task, the final layers necessitated modification. The acquired dataset was imported and partitioned into training (70%) and testing (30%) subsets. To mitigate overfitting and promote generalizability, data augmentation strategies were employed: Random reflection along the X-axis, Random reflection within the range of  $-90^{\circ}$  to  $90^{\circ}$  and Random rescaling between 1 and 2. These data augmentation techniques enabled the network to develop robust feature representations. The network's performance was assessed to achieve zero error. Training parameters, including epoch, initial learning rate and frequency, were meticulously optimized. The network underwent training until it attained a classification accuracy of 90%, this is shown in Figure 5 and surpasses the threshold of 80% established by Suhaili and Abdullah (2013). Following successful training, a novel image was imported and normalized, facilitating classification by the trained network.

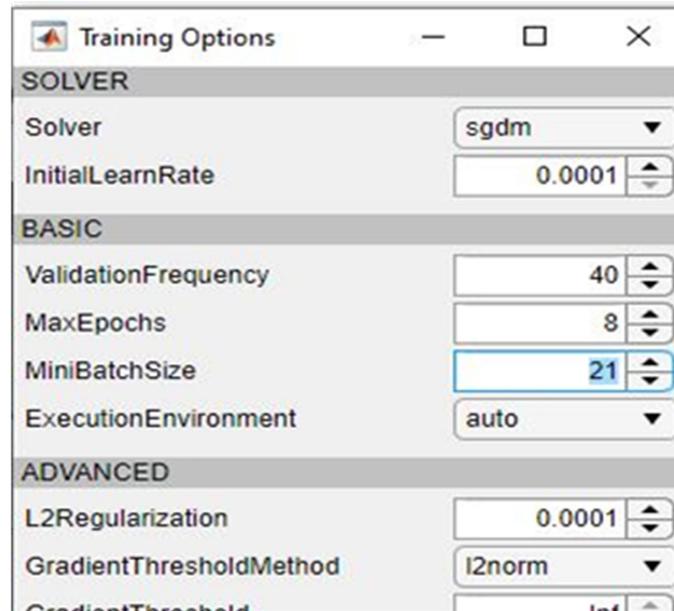


Figure 5: Sample Training Option

### 3.2 Feature Extraction With SPCNN-SVM

The image feature extraction and classification pipeline consisted of two primary stages: Feature extraction: Utilizing the pretrained SPCNN model to extract discriminative features from the image dataset; Classification: Employing SVM to classify the extracted features. SPCNN was loaded, and the image dataset was imported and normalized. Network analysis was performed to ensure optimal performance. Subsequently, SPCNN extracted relevant features, which were then passed to SVM for classification.

The SPCNN facilitated the construction of a hierarchical representation of input images, wherein deep layers integrated lower-level features to form higher-level features. Feature extraction was achieved by applying the ‘activations’ function to the output of pool10, after image augmentation. The extracted features from the training images were employed as predictor variables for SVM-based classification. Following the network analysis, a novel test image underwent: Loading and normalization; Prediction and classification using the trained SVM model.

### 3.3 Evaluation Measures and Validation

The performance of SPCNN and SPCNN-SVM was evaluated using Accuracy (A) in Equation 2, this is the proportion of correctly classified instances which measured the network’s predictive accuracy; Training Loss (T) in Equation 3, this is the proportion of misclassified instances which indicated the network’s training error; Elapsed Time (E) in Equation 4 which is the computational time required for training and testing. These evaluation metrics provided a comprehensive assessment of the network’s performance. The effectiveness of the trained network was validated using live images of cassava leaves, encompassing both healthy and diseased samples. The images were input into the network, which generated class labels and accuracy scores for SPCNN; and class labels only for SPCNN-SVM. This validation process evaluated the potential for SPCNN deployment in a production setting, ensuring its reliability and effectiveness.

$$A = \frac{TP+T}{(TP+TN+FP+FN)} \quad (2)$$

$$L = 1 - \frac{TP+TN}{(TP+TN+FP+FN)} \quad (3)$$

$$E = \text{No of cycles} \times \text{average time} \quad (4)$$

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

## 4. RESULTS AND DISCUSSION

### 4.1 SPCNN Result

Figure 6 displays the initial Transfer Learning (TL) training outcomes, achieving an accuracy of 92.22% with a computational time of 17 mins 39 secs. Subsequent transfer learning experiments, employing varied parameters, are detailed in Table 1. Figure 7 presents the validation results of the SPCNN model, showcasing its effectiveness in predicting cassava leaf diseases: Image 1: Healthy (93.2% accuracy); Image 2: Witches' broom disease (99.2% accuracy); and Image 3: Spot disease (61.9% accuracy).



Figure 6: Training Result for First TL



Figure 7. SPCNN Validation

Table 1. Different Training Options and Accuracies

Frequency	Accuracy	Training Loss	Elapsed Time
70	91.67%	08.33%	18m31s
60	90.56%	09.44%	20m29s
56	90.28%	09.72%	20m42s
40	92.22%	07.78%	17m 39s
35	86.94%	13.06%	20m 03s
30	90.83%	09.17%	19m 32s
15	80.28%	19.72%	22m04s

#### 4.2 SPCNN-SVM Results

The zero-error analysis for the first feature extraction process is illustrated in Figure 8. In this analysis, SPCNN was employed to extract relevant image features, which were then classified using a multiclass SVM. The validation result of this process is presented in Figure 9. Notably, the SPCNN-SVM model demonstrated excellent predictive accuracy, correctly classifying the input images as having spot, blight and witches' broom diseases.

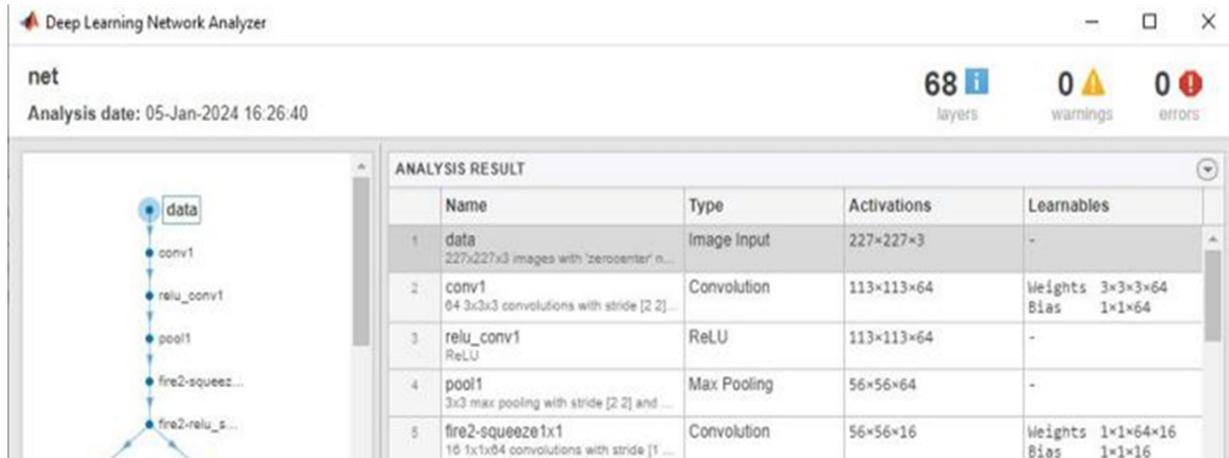


Figure 8. Feature extraction zero-error analysis



Figure 9. SPCNN-SVM validations

#### 4.3 Comparison OF SPCNN and SPCNN-SVM Results

A comparative analysis of the performance metrics revealed that the SPCNN-SVM model outperformed SPCNN in terms of accuracy and computational efficiency. Specifically, SPCNN-SVM achieved higher accuracies and required less time to complete training. However, SPCNN offered several advantages including consistent feedback throughout the training process, detailed prediction outputs including prediction values and percentage accuracies, as well as enhanced transparency, allowing for real-time monitoring of training progress. In contrast, SPCNN-SVM only provided predicted results. Table 2 summarizes the comparative results of SPCNN and SPCNN-SVM, highlighting the superior accuracy of SPCNN-SVM. Nonetheless, SPCNN's output provided additional valuable information, such as accuracy scores at various epochs and time utilization per epoch. Further analysis revealed that SPCNN's performance improved with increasing frequency, with higher frequencies yielding higher accuracies. Notably, reducing the frequency to 15 resulted in the lowest accuracy of 80.28%. The optimal accuracy achieved by SPCNN-SVM was 98.89%, the overall highest accuracy.





6. Kamilaris A. and F. X. Prenafeta-Boldu (2018): "A review of the use of convolutional neural networks in agriculture". *The Journal of Agricultural Science*, Cambridge University press, 156(3):312-322doi:<https://doi.org/10.1017/S00218596180004>.
7. Mingyue S., Peitong H, Yanqi Z., Shuo Z., Ning Z. and Jianhua Z. (2023): "Identification Method of Cotton Leaf Diseases Based on Bilinear Coordinate Attention Enhancement Module". *Agronomy Journals*, 13(1):8888/10.3390/agronomy13010088.
8. Sachin D. Khirade, Patil A. (2015): "Plant Disease Detection Using Image Processing". 2015 International Conference on Computing Communication Control and Automation. 17805229. Doi:10.1109/ICCUBEA.2015.153
9. Sarvary, S., Ficke, A., Aubertot, J.-N., Hollier, C. (2012),"Crop losses due to diseases and their implications for global food production losses and food security", *Food Security*, 4:519–537.
10. Suhaili B. K. and N. Abdullah. (2013). A Survey Paper on Plant Disease Identification Using Artificial Neural Network. 04(07):13
11. Zhang, Y.-D., Dong, Z., Chen, X., Jia, W., Du, S., Muhammad, K., Wang, S.-H. (2019), "Image-based fruit category classification by 13-layer deep convolutional neural network and data augmentation", *Multimedia Tools and Applications*, 78:3613–3632.