International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Tomato Crop Yield Prediction in Indoor Environment with A Novel ABC Enhanced CNN with SDL Architecture

[1]Mrs. D. Radha, [2]Dr. S. Prasanna

Submitted: 01/10/2023 **Revised**: 22/11/2023 **Accepted**: 02/12/2023

Abstract: More over half of India's population relies on agriculture for their livelihood. Population growth means more people need to eat, thus growing food inside is essential. Tomatoes is one of the world's most extensively grown vegetable crops. It's on par with the fourth-best veggies in the world. During the development of a tomato, it is susceptible to a wide range of illnesses and pests. Reduced yields or even crop loss might occur if management measures are delayed. Accurately identifying the source and impact of crop output is crucial for figuring out how to manage diseases and pests efficiently and helping vegetable growers boost tomato yield. Multiple variables, including genetics, environment, and their interactions, impact crop yield, making it a highly complicated characteristic. Predicting yields with any degree of precision calls for first gaining a deep grasp of the functional connection between yield and various interaction components. In this research, we use a dataset from kaggle to forecast the yield of tomatoes grown indoors using a hybrid approach: the ABC-CNN classification algorithm. The prediction method makes use of the Convolutional Neural Network (CNN) model and the Artificial Bee Colony (ABC) algorithm with the Adam optimizer. The results demonstrate that our suggested technique outperforms competing algorithms in terms of accuracy and efficiency.

Keyword: *Tomato, Yield prediction, ABC, CNN*

1. Introduction

The importance of growing vegetables in greenhouses for human consumption is rising. It has become an important part of factory farming, which supplies much of the world's vegetables. Many contemporary greenhouses now employ clever control algorithms to completely automate the ventilation, auxiliary lighting equipment, and sunshades in order to significantly increase productivity while saving energy. The fast evolution of today's greenhouses has made it imperative that greenhouse plants be digitized and visualized for use in indoor farming. The endeavor may be difficult because of the crop's complex surroundings and structure, but new methodologies and sensors have been driving the study, making it an intriguing and promising issue among agriculturists, engineers, and botanists. Practical applications of plant digitization and visualization research include (1) accelerating the development of intelligent greenhouse control technology; (2) facilitating the study of plant physiology and ecology; (3) warning of plant diseases and pests earlier or even in advance; and (4) aiding in robotic harvesting.

Due to the large proportion of the Indian population that works in agriculture (66.5%), safeguarding crops against pests and diseases has become an urgent issue. The specific disease's onset is strongly influenced by its environment and climate. Existing pathogens in the soil, air, water, and agricultural detritus lay dormant until they detect favorable circumstances. As a result, farmers need to keep a close eye on their fields at all times in order to detect the first signs of illness in their crops. As soon as visible symptoms manifest, conventional sample collection and analysis begin. Conventional diagnostic procedures need laboratory equipment and trained staff to perform a variety of immunological tests. Though effective, this technique has practical constraints due to its high cost, substantial labor need, and, most significantly, lengthy execution time. We need a simple, rapid, user-friendly, and cost-effective system with little human interaction to take their place, so that we can keep an eye on things round-the-clock. The demand for automation is especially pressing in plant pathology, where resources are few and trained professionals are in short supply. Several scientists have shown an interest in studying automated plant disease diagnosis throughout the years. Machine learning, ANN, and CNN are just a few examples of the cutting-edge computational technologies that have paved the way for the development of non-destructive algorithms and methods [2].

Damage from plant pests and diseases may destroy whole plants or large sections of them, reducing agricultural yields and perhaps triggering a food shortage. Not everyone in the nation is well-informed about how to deal with pests or prevent the spread of illness. New techniques have been developed to lessen the processing required after harvest, improve agricultural sustainability, and

[1]*Research Scholar, Vels Institute of Science, Technology and Advanced Studies(VISTAS)*
radha.aasc@gmail.com
[2]*Professor Department of Computer Applications Vels Institute of Science, Technology and Advanced Studies(VISTAS)*
prasanna.scs@velsuniv.ac.in

boost output. Thermography, mass spectrometry, gas chromatography, polymerase chain reaction, and hyper spectral techniques are only few of the methods utilized to diagnose the illnesses. The preceding methods need extensive processing time and may be rather expensive. Disease detection methods now in use include both mobile and server-based approaches. Accurate and efficient illness detection is achieved via the use of these methods, which feature, among other things, cameras with high resolution, processors with high configuration and strong performance, and high-end built-in accessories. Disease detection, identification, and classification are all areas where cutting-edge methods like machine learning and deep learning are being used. Fuzzy logic, neural networks, and deep neural networks are only some of the machine learning methods being studied for their potential to identify and categorize plant illnesses [3].

Tomatoes are a versatile produce that may be used in a wide range of dishes and beverages, from savory to sweet. Although India produces second-most tomatoes worldwide, its output per acre is quite low compared to that of other nations. According to the Indian Institute of Horticultural Research, bacterial diseases (wilt, early blight) or the leaf curl virus may wipe out 70-100% of a tomato crop in India. Many research facilities are investing significant effort and money into developing hybrid disease-resistant cultivars as a possible solution. Predicting illnesses early on and administering the right treatment at the right time may save a lot of output, and such improvements are only possible because of technological progress [4]. Tomato output is severely impacted by the prevalence of tomato pests and diseases in various geographical locations. A decrease in production or total crop failure might result from delayed management measures. In order to grow vegetables without contributing to pollution, it is essential to take measures to avoid disease and pests. When plants have matured to the point that pests and illnesses are showing symptoms, it is also a passive cure, even if humans can accurately diagnose and treat them. Although this is important, modern agricultural chemical and pesticide applications often have a negative impact on the environment, leading to problems like high levels of pesticide residue in fruits and vegetables and the development of pest and disease resistance that complicates efforts to protect crops from harm. Therefore, it is crucial to anticipate the emergence of pests and illnesses and take precautions accordingly. Tomato studies illustrate how easily a plant may be affected by pests and illnesses. Accurate diagnosis of illnesses and insect pests is crucial for successful treatment of these problems and for assisting vegetable growers in increasing tomato yields. Scientists and technicians have their greatest hurdle when trying to pinpoint the source of tomato illnesses and insect pests [5].

The tobacco mosaic virus has a global spread and is the most invasive and destructive plant virus ever recorded. Tobacco is a major economic crop in our nation, but the prevalence of tobacco mosaic disease has drastically cut down on both production and quality. Given that plants lack a fully functional immune system, tobacco mosaic virus is notoriously difficult to eradicate once it establishes itself. Upon infection, the plant's leaves may develop mosaic symptoms or even malformations, and the growth may become chronically ill. Many researchers are interested in studying viruses, and with the advent of machine learning techniques, many have begun using these methods to their work [6].

Experts may learn a lot just by looking for signs of plant disease with their own two eyes. However, this strategy requires the constant presence of professionals, which may be costly and time-consuming, especially for expansive farms and outlying areas. We need to look for a rapid, automated, less costly, and accurate way to identify illness since proper and early detection of diseases is highly crucial. These goals may be accomplished through image processing. Agricultural applications of image processing include the following: (1) detecting diseased leaf, stem, or fruit; (2) quantifying affected area by disease; (3) discovering the form of affected area; (4) identifying the color of affected area; and (5) establishing the size and shape of fruits.In recent years, automated plant disease diagnosis and classification in leaf images has been greatly aided by image processing methods using machine learning (ML) algorithms. They [8] illustrate the system architecture of a computerized method for diagnosing and categorizing plant diseases. It usually consists of a two-stage process. The first stage is comprised of image processing procedures, including image acquisition (a technique for capturing images of the diseased portions of the plant leaf using an RGB camera), image pre-processing (a technique for removing noises in the captured image using filters), image segmentation (a technique for separating the infected area from the healthy one in a given image), and feature extraction (a technique for configuring the derived values from the segmented image). In the second stage, leaves are analyzed to determine whether they are healthy or contaminated by a machine learning (ML) algorithm [8].

One of the most pressing issues in precision agriculture is the development of methods for disease detection using just photographs of plant leaves. Technology advancements in computers, image processing, and the newest findings in Neural Network research hold significant promise for enhancing plant growth as well as safety methods. Many popular methods for diagnosing and categorizing plant diseases are based on artificial intelligence (AI). Neural Networks, Logistic Regressions, Decision Trees, Support Vector Machines (SVM), k Nearest Neighbors (k-NN), Na ve Bayes, and Deep Convolutional Neural Networks (Deep CNN) are some of the most widely used AI methods. In order to attain high accuracy in plant disease diagnosis, recent reports from several researchers have shown that Deep Learning is the best method. Many studies [9] have shown promising outcomes in illness categorization when researchers use transfer learning on models that have been pre-trained from other domains. Using data from kaggle, this research proposes a novel ABC-CNN classification method for forecasting the production of greenhouse-grown tomatoes. Using a Convolutional Neural Network (CNN) model and the adam optimizer, the Artificial Bee Colony (ABC) method can make predictions.

2. Related Works

The deep convolutional neural network (CNN) used in [10] was created by Muhammad E. H. Chowdhury and colleagues using the Efficient Net CNN paradigm. The model was taught to recognize differences between good and diseased photos of tomato leaves. Using the most widely-used public Plant Village dataset, our approach beats several recent deep learning algorithms [60,61]. Image segmentation using the Modified U-net was found to be most effective at separating leaves from backgrounds, whereas discriminative features were most effectively extracted using EfficientNet-B7. In addition, the networks' performance often flourished as more parameters were used during training. Using the trained models, plant diseases may be automatically detected early on. Early illness diagnosis through visual examination often requires years of experience as well as training for experts, but our technique is accessible to anybody with a basic understanding of epidemiology. The network will automatically begin gathering data from the user's visual camera as soon as they log in, and they will be notified promptly so they may take appropriate action. As a result, precautions may be done before it's too late. The use of cutting-edge tools like cellphones, drone cameras, & robotic platforms may make this study useful for early and automated disease identification in tomato crops. Better crop yields may be ensured with the help of the suggested framework and a system of feedback that provides helpful ideas, treatments, disease management, as well as control measures. The authors plan to do further work to verify the effectiveness of the suggested approach in a real-time application, using microcontrollers equipped with cameras to monitor the results.

Tobacco disease injected with TMV may be quickly identified using hyperspectral imaging within the Vis/NIR spectral range (380-1023 nm), as shown by the work of Hongyan Zhu et al. [11]. Reflectance spectra were correlated with illness progression using a variable selection approach and machine learning models. After wavelengths were selected using SPA, the total number was cut by more than 98%, greatly reducing the complexity of the calibration models or the computations required to use them. The majority of our machine learning models have a classification accuracy of 85% or above. ELM and BPNN models also achieved detection and classification rates of 98.33% and 96.67%, respectively, when it came to healthy and sick tobacco leaves (2 DPI, 4 DPI, and 6 DPI). The selection of EWs did not significantly affect the calculation time of the different machine-learning approaches. It was suggested that these extremely efficient strategies be put into use. The LS-SVM, ELM, and BPNN models outperformed competing machine learning models, expanding researchers' options.

Using the newly developed Inception Net CNN design, Muhammad Shoaib et al. [12] demonstrated the results of a CNN. Successfully classifying images of tomato leaves as either healthy or unhealthy, the CNN model proved useful. These findings, acquired using the publicly accessible benchmark dataset Plant Village (Hughes & Salathe, 2015), show that our model performs better than a variety of state-of-the-art deep learning approaches. Modified U-net outperformed competing architectures when it came to extracting leaf pictures from their contexts. When compared to other architectures, InceptionNet1 was also the most effective in omitting low-importance details from images. The suggested framework may be used to improve disease management, crop recommendations, control measures, and harvests by including a feedback mechanism.

According to research by Parul Sharma et al. [13], most deep learning models of autonomous illness identification have poor results when used on unseen photos from the actual world. In this study, we demonstrate that training a CNN model using just segmented and annotated pictures is possible. Using segmented pictures (S-CNN) to train the same CNN model as full images (FCNN) improves model performance on independent information from 42.3% to 98.6%. Eighty-two percent of the test dataset also shown an increase in confidence during quantitative examination of self-classification.

According to Ahmed Ali Gomaa, et al. [14] Using the CNN algorithm for classification and prediction, DL was utilized for early prediction to identify illnesses at various stages of plant development. In this case, a TMV-infected tomato served as a model and correctly diagnosed TMV infection 97% of the time. The GANs were able to boost both the data size and the prediction accuracy rate from the original data by a factor of 98%. It was shown that the second growth stage group was the most susceptible to viral infection across all plant life cycles. Since the stages of unhealthy development (healthy -> first infection -> unhealthy) were also determined in this article, it follows that the data obtained show the age group most vulnerable to unhealthy. The study's findings have been confirmed by applying them to a dataset of actual data gathered by hand from one of Egypt's farms. Rapid advancements in DL models, transfer learning approaches, and CNN frameworks mean that several DL models for early detection as well as classification of plant diseases will be implemented in future studies. In order to achieve the best possible prediction accuracy, we will test our model on a larger real-time dataset of TMV-infected tomato plants and other significant plant-disease systems.

In [15], Lili Li et al. introduced readers to the fundamentals of deep learning and provided a thorough analysis of current research on the use of deep learning to the problem of plant leaf disease recognition. Assuming a sufficient amount of training data is made accessible, deep learning methods may accurately identify plant leaf diseases. Large datasets with high variability, data augmentation, transfer learning, and visualization of CNN activation maps have all been discussed as means of enhancing classification accuracy; small-sample plant-leaf disease detection; and hyper-spectral imaging for early disease detection in plants. However, it is not without flaws either. While most DL frameworks in the literature perform well on their own datasets, they often struggle to perform well on different datasets, indicating that the model is not very resilient. As a result, robuster DL models are required to accommodate the various illness datasets.

In their discussion of plant diseases, Hardikkumar S. Jayswal et al. [16] covered topics such as their definitions, the numerous methods used to identify them, how they are categorized, and how they compare to one another. There are several pathogens present in the plant agricultural industry. Bacterial, viral, and fungal illnesses are the three broad groups into which all of these conditions fall. In this review, we provide the conventional approach, which is comprised of known image processing algorithms for disease detection in plants (as illustrated in Fig. 4). They found that k-means segmentation or Hue Based segmentation were the most popular methods for segmentation, while a wide variety of machine learning classification algorithms (support vector machines, artificial neural networks, decision trees, naive bayes, probabilistic graphical models, and boosted decision trees) were used for classification. In terms of accuracy, the SVM & NN are the most used classification algorithms.

Vijaya Mishra et al. [17] The most common and devastating diseases that attack tomato plants have been covered. Tomato leaves, both healthy and flawed, have been analyzed statistically to determine their characteristics. The statistical aspects of healthy and defective tomato leaves are discovered to be quite different after an examination of the acquired data on leaf characteristics. Defective leaf photos are evaluated for intensity distribution based on disease category and associated histogram. Analyzing the statistical properties of leaves reveals a striking discrepancy in feature value. The histogram representation makes it possible to extract color information that aids in locating specific plant diseases. To prevent further damage to the tomato crop and its associated costs, an early disease prediction system based on characteristics extracted from the leaves may be developed.

Jie Sun et al. [18] Accurate early yield forecast is crucial for managing harvests, organizing agricultural markets, and obtaining crop insurance. In this study, we present a GEE-based CNN-LSTM model for predicting CONUS soybean production at the county level during the growing season. The findings provide for the first time proof that the suggested CNN-LSTM model's prediction performance was demonstrated to be the best compared with the CNN or LSTM from 2011 to 2015. Using remote sensing data, machine learning techniques like Deep Learning's Convolutional Neural Networks and Long Short-Term Memories have made significant strides in this area. New research in this field suggests that LSTM may disclose phenological traits and that CNN can investigate additional spatial variables, both of which are crucial to crop output prediction.

According to Muhammad SobriRamli et al [19] research, image-based Machine Learning architecture may be used to identify a number of common illnesses affecting the tomato plant. This list includes things like late blight, early blight, bacterial spot, leaf spot, yellow leaf curl virus, leaf mold, mosaic virus, and target spot. The majority of the proposed solutions made use of the CNN-based architecture, which has shown to deliver very accurate detection. While other architectures can interpret the same plant leaf samples with some degree of accuracy, GoogLeNet and AlexNet continue to lead the pack.

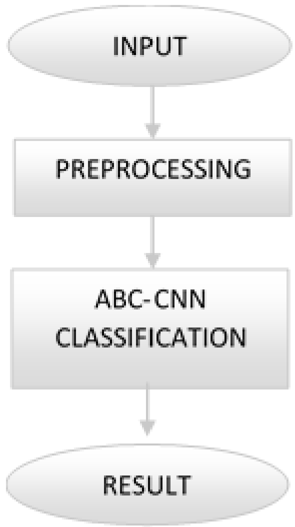
Summary:

- There is a detrimental effect on agricultural output due to the prevalence of plant diseases. Increased food insecurity may result from delayed diagnosis of plant diseases. The management and decision-making of agricultural output relies heavily on early identification, which is the foundation for successful control and prevention of plant diseases.

- Increased yield, quality, local marketing, and international export competitiveness can be achieved through early plant protection achieved through the development of a reliable and accurate digital and computer-based early detection and warning system for plant pest-infestations and microbial disease-infections.
- Many bacterial & fungal illnesses may infect or attack tomatoes. Because many illnesses are so difficult to contain, early diagnosis is crucial.
- By giving a huge number of real-time photos, the CNN-based network may be taught to detect plant illnesses.
- Plant illness may be predicted using photos of plant leaves from both healthy and sick plants, as well as from a future training model.

3. Proposed Method

One of the causes of decreased agricultural output is the presence of pest insects and illnesses, both of which may be detected and dealt with with the use of modern information technology. Using enough information may boost productivity by decreasing waste and increasing production. As a result, deep learning has been the subject of extensive study by academics and professionals throughout the world. When used to the categorization of vast amounts of data, deep learning dramatically lightens the strain and speeds up the attribution process. The most distinguishing features of deep learning are its complex network topology and large data samples. Our crop production prediction model has been given substantial technological help with the advent of deep learning technology. As can be seen in Figure 1, this work makes use of a hybrid ABC-CNN model to forecast the yield of tomatoes grown indoors.



i) Input data:

The dataset, which we downloaded from Kaggle, is expected to provide data on a wide range of possible influences on tomato crop quality and production. The features are Index, AssimLight, BleckScr, CO2eir, Cum_ir, EC_drein_PC, EnScrHumDef, PipeGrow, PipeLow, Rhair, Tair, TotPAR, TotPARLemps, VentLee, Ventwind, pH_drein_PC, weter_sup, Terget as shown in table 1. The data value for TotPAR is 0, TotPARLemps is 0, VentLee is 1 and Ventwind is 0.

Table 1. Some of the input data values got from dataset

Index	AssimLight	BleckScr	CO2eir	Cum_ir	EC_drein_PC	EnScr	HumDef	PipeGrow	PipeLow	Rhair	Tair	weter_sup	Terget
0	100	35	509	31.6	0.3	96	8.8	0	49.9	51.9	21	263	class A
1	100	85	484	31.8	0.3	96	9.2	0	48.5	51.3	21.5	265	class A
2	100	96	475	31.8	0.3	96	9.1	0	46.8	52.2	21.6	265	class A

3	100	96	501	32	0.3	96	8.5	0	45.2	54.6	21.3	267	class A
4	100	96	487	32	0.3	96	8.5	0	43.8	54.4	21.4	267	class A
5	100	96	472	32.3	0.3	96	8.5	0	42.4	54.3	21.3	269	class A
6	100	96	491	32.3	0.3	96	8.5	0	41	54.3	21.2	269	class A
7	100	96	487	32.5	0.3	96	8.4	0	41.5	54.6	21.1	271	class A
8	100	96	490	32.5	0.3	96	8.2	0	43.2	54.7	20.9	271	class A
9	100	96	506	32.8	0.3	96	8.3	0	44.4	54.8	21	273	class A

Figure2. Dataset

ii) Preprocessing:

- NaN Value Checking: To make sure the dataset was full and usable for training, we looked for missing (NaN) values.
- Data Transformation via Label Encoding: To prepare the data for use in a deep learning model, we used label encoding to transform categorical variables into numbers.
- Data Scaling via Standard Scaler: Figure 2 shows the benefits of using standard scaled to normalize a dataset's numerical features, which ensures that features are in comparable scales and promotes better model convergence.

```
AssimLight       0
BlackScr         0
CO2air           0
Cum_irr          0
EC_drain_PC      0
EnScr            0
HumDef           0
PipeGrow         0
PipeLow          0
Rhair            0
Tair             0
Tot_PAR          0
Tot_PAR_Lamps    0
VentLee          0
Ventwind         0
pH_drain_PC      0
water_sup        0
Target           0
dtype: int64
```

Fig 2. Data Cleaning – Null Value Checking Result

Train-Test Split: We split the dataset into a training set and a test set after preprocessing it. When developing a deep learning model, data from the training set is utilized during development, while data from the test set is retained in isolation for testing purposes.

Training the Model: The model was educated with help from the training data. The model is trained to recognize patterns in the data and generate predictions based on those patterns.

Model Prediction: After the model was trained, we utilized it to make predictions about the testing data's tomato crop quality yield.

iii) Classification Method

a) ABC

For multivariable numerical optimization problems, Karaboga presented the artificial bee colony method based on swarm intelligence [35]. This method is inspired by the swarm intelligence and self-organization paradigm seen in bee colonies.

There are three essential components to the ABC algorithm's artificial bee colony: food sources, working bees (sometimes called leading bees), and idle bees. Both observers and scouts may be found among the unemployed bees. There are three distinct patterns of behavior that result from the interplay of these factors: the presence of food supplies that draw bees, the pursuit of food sources by bees, and the abandonment of food sources by bees.

Bees that are "employed" have been trained to associate with a certain food supply, which they are actively using. Bees gather in the hive to watch each other while they forage for food near an established food source. The crowd gathers in the dancing area while waiting to choose their meal. The "roulette wheel selection" approach is used to choose which food source will be used. When a worker bee leaves a feeding location, it becomes a scout & forages at random for food in the area around the nest. The algorithm forms a population, keeps the good members while getting rid of the bad ones, and gets closer and closer to the global optimum solution with each iteration [21].

Based on population dynamics and honeybee foraging behavior, ABC is a Meta heuristic optimization technique. The number of solutions, which represents the available food sources, the maximum cycle number, which represents the maximum number of generations, and the limit, which symbolizes the removal of the available food source, are the only three control parameters in ABC. ABC is related to selection and population growth. Population refers to the act of exploring the different areas inside the search space, while selection refers to the process of ensuring the prior experiences are used. The ABC framework consists of four stages: 1. Second Stage: Worked Bees Phase of the bees who just watch Four, the scouting-bee-like stage. It all comes down to how much nectar the source produces and where it is located. There are as many workers as there are food sources, or solutions, and each food source has a dedicated worker. First, scout bees begin the population and control parameters, as well as the food sources (solutions), during the initiation phase. If a solution is identified in the first phase, the employed bee phase, the fitness values are evaluated using a roulette wheel (greedy selection) approach. Information regarding food sources is shared between worker bees and spectator bees in the hive's dance area. When a scout bee's maximum number of attempts to enhance something (called limit) has been reached, it gives up and moves on to a new food source. Eventually, the scout bee will begin its quest for new food sources at random [20].

Algorithm1. ABC algorithm [27]

1. Set the starting locations of food sources Repeat
2. Putting workers bees in charge of finding food sources
3. Probability values for use in probabilistic selection, which may be calculated
4. Probability-based positioning choice for food sources by foraging bees
5. Onlooker bees are swarming that area because of the food.
6. Giving up on less likely sources and starting new food production close to the previous one.
7. Until the maximum cycle count is reached or the required error is achieved

Food sources: Profitability of food sources are represented as fitness of a solution, which is simulated by the location of the solution to the optimization problem.

Unemployed foragers: Both scouts and observers fall within this category. Finding and using new food sources is part of their job.

Employed foragers: There are as many of them as there are places to get food. The worker bees remember where the food comes from and pass that information along with a certain degree of probability. If the business's food supply runs out, the worker bee will be converted into a scout.

In order to determine which food source will provide the most profit, the observer may watch a number of dances performed by worker bees and use this knowledge. The observer bee calculates its profit from the available food sources based on their probabilities. As a result, the attractiveness of a food supply directly affects the number of recruits taken in [27].

b) Hybrid ABC-CNN algorithm

In order to effectively locate important characteristics within the data, a CNN uses layers such as convolution layers, pooling layers, as fully connected layers, which are not included in traditional neural network techniques.

Feature extraction is handled by the convolution operation or activation function that make up the convolution layer. The convolutional algorithm is composed of a filter and a feature map. Feature maps are the result of filters, which are collections of weights applied to the input. Down-sampling is performed with the use of a pooling procedure, which aids in the detection of characteristics. In order to introduce nonlinearity into the output, the results are fed into a nonlinear activation function. The network may learn the mapping between a features or the target by adding fully connected (FC) layers following convolution layers [28].

CNNs can handle data in a variety of array formats, from one-dimensional signals and sequences to two-dimensional pictures and three-dimensional video. Multiple convolutional and pooling layers, followed by a small number of fully connected (FC) layers, make up the typical CNN model. The filters, padding, and stride of a CNN are some of its design parameters. We convolve the input data with a filter, which is a matrix of weights. To maintain the original dimensions of the input space, padding is used. The filter's distance traveled is known as the stride [23].

The Adam [15] algorithm is a method for optimizing the pace at which one is learning. The deep learning paradigm inspired the development of this optimization technique. The algorithm's main contribution is the identification of unique adaptive learning rates for different parameters. The adaptive moment estimation process inspired the algorithm's name. Gradient estimates at the first and second moments are used to fine-tune the learning rate of each and every weight in a deep neural network. The first instant is the mean, while the second is the variance. The moments in each batch are estimated using exponential moving averages in the Adam technique. The following mathematical formulae may be understood in light of Adam's update rule:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)h_t \tag{1}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)h_t^2 \tag{2}$$

In this equation, the variables m and v represent moving averages, h the gradient on the current batch, t the number of iterations, and β1 and β2 hyper-parameters of the method. Moving averages of the gradient and the squared gradient are shown in Equations (1) and (2), respectively [22].

When training a CNN model, an Adam optimizer with a power-exponential learning rate is used to speed up the optimization process of the standard Adam optimizer, with the power-exponential learning rate determining the iteration direction and step size. Based on the previous stage's learning rate and the gradient connection between that stage and the present stage, the power-exponential learning rate may be adaptively modified. In order to fine-tune the correction factor for adaptive adjustment, the prior gradient value is employed. As a result, the learning rate can be fine-tuned within a narrow range, the parameters of each iteration remain relatively constant, and the learning step is chosen based on the appropriate gradient value to alter the network model's convergence performance and guarantee the network's stability and efficacy [26].

CNNs, or convolutional neural networks, are deep learning models optimized for processing grid-based information. These may be pictures or rows of data with several columns. The term "deep learning" is used to describe complex models with many levels. If a model contains an input layer, a hidden layer, and an output layer, it is said to be deep. Artificial neural networks borrow their operating principle from biological neural networks, which use neurons as its fundamental building pieces. CNNs have a number of advantages over more conventional feedforward neural networks when it comes to locating important patterns in data.

Salient features are those that have the most illustrative power regarding the data-generating process, and they are the ones that are extracted by a CNN's convolutional layers from input pictures. Successful mapping to a target value is necessary before the learnt features can be used in a regression or classification operation. To do this, normally fully connected (FC) layers are added after the convolutional layers. The word "fully connected" refers to the idea that all neurons (or units) in the preceding layer are linked to all neurons (or units) in the current layer. Adding more FC layers improves the network's ability to discover the feature-to-target mapping. It also makes optimization more difficult since the number of connections in FC layers rises exponentially with their depth [25].

Algorithm2. Hybrid ABC-CNN model

1. Input: The population size, a maximum generation number, the picture dataset for classification, the number of food sources (N), and the maximum number of iterations (g) are all parameters that have been specified.
2. while t < maximum generation size
3. Determine each person's Pt fitness level using the suggested acceleration variables;
4. Qt ← Propose a mutation and use crossover operators to produce children from the chosen parents ;

5. Pt+1 ← Environmental selection from Pt ∪Qt;
6. t ← t + 1;
7. end
8. Send back the player who, in Pt., has shown the most fitness.
9. Initiating the ABC Parameters
10. Build the first hive and colony of bees.
11. Changes in individual bees' fitness levels over time
12. Repeat
13. N=0
14. Repeat
15. Use greedy selection if possible
16. Determine probabilities with the help of Equation 2
17. Replace the bees at positions vi & xi with fresh observers.
18. Do for all onlooker bees
19. Perform greedy selection
20. If onlooker fitness probability is < employed fitness probability
21. The use of probability instead of the onlooker value
22. END
23. END
24. If BEST onlooker fitness < Best Fitness
25. Replace onlooker fitness with best fitness
26. END
27. N=N+1
28. Until N= Employed Bee Number
29. Using Equation 1 identify the abandoned solution
30. If the scout solution is more effective, it should be implemented instead of the current method.
31. Iterate until the maximum number of iterations is reached (31) and then return the band combination of the food supply a ti that results in the greatest fitness Fi in the population A.

- The suggested model's design is made up of five interconnected levels. The first two layers have 256 neurons each, the third and fourth layers have 128 neurons apiece, and the fifth and final layer has 64 neurons. Each of these thick layers has a dropout rate of 0.1 added to it to avoid over fitting.
- The model is finished off with an output layer that uses a sigmoid activation function. To best accomplish the job at hand and provide the required prediction, an activation function is selected.
- In order to get optimal results when training, the ADAM optimizer is used. This optimizer is well-known for its efficacy and efficiency in adjusting training rates to meet the needs of each parameter.
- A binary cross-entropy loss functions is used for forecasting. This loss function is well-suited for binary classification problems since it quantifies the degree to which predictions deviate from reality.

In conclusion, the suggested model architecture has many thick layers, each of which is provided with dropout regularization. The sigmoid activation is used in the last layer that is the output. Training for accurate prediction is accomplished with the help of the ADAM optimizer and a binary cross-entropy loss function.

iv) Results

About 2000 items with 17 characteristics remain after filtering procedure in the dataset utilized for the operation. There are now 1400 records for training & 600 for testing.

Data Visualization

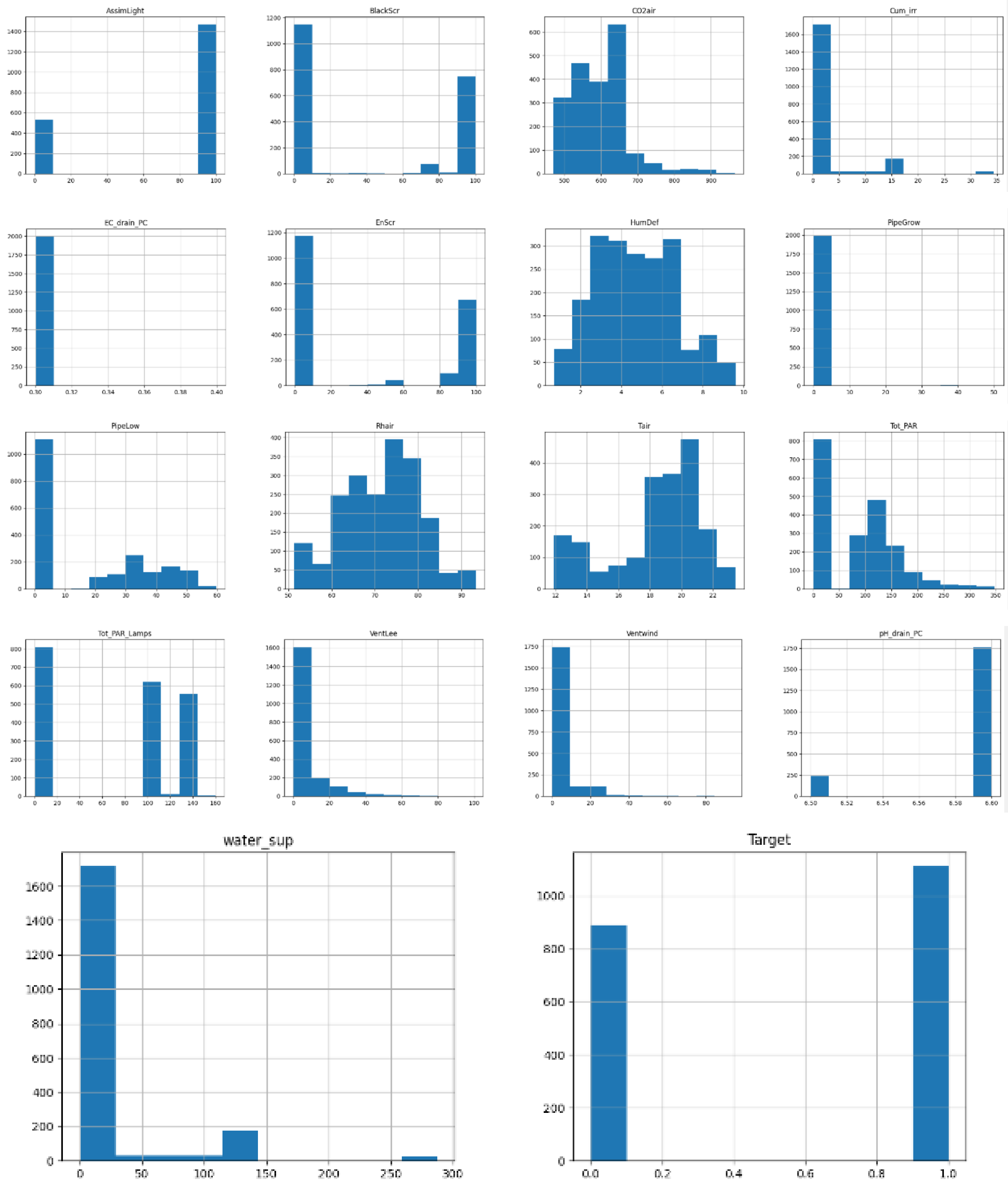


Fig 3. Histogram plot for all variables

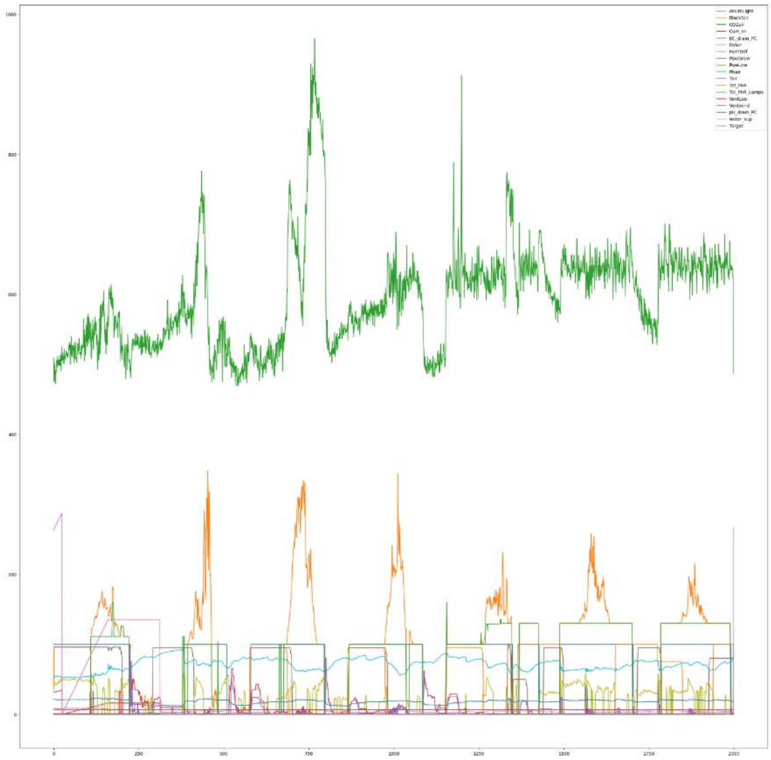


Fig 4. Data distribution plot

MODULE 3 – Data Splitting

Data Splitting

```
training set size: (1400, 17), testing set size: (600, 17)
```

MODULE 4 – Model building

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 256)               4608

 dropout (Dropout)           (None, 256)               0

 dense_1 (Dense)             (None, 256)               65792

 dropout_1 (Dropout)         (None, 256)               0

 dense_2 (Dense)             (None, 128)               32896

 dropout_2 (Dropout)         (None, 128)               0

 dense_3 (Dense)             (None, 128)               16512

 dropout_3 (Dropout)         (None, 128)               0

 dense_4 (Dense)             (None, 64)                8256

 dropout_4 (Dropout)         (None, 64)                0

 dense_5 (Dense)             (None, 1)                 65

=================================================================
Total params: 128,129
Trainable params: 128,129
Non-trainable params: 0
```

Fig 5. Proposed Model Structure (Sequential Deep Learning Model)

Training Process

```
Epoch 1/10
44/44 [==============================] - 2s 7ms/step - loss: 0.3090 - accuracy: 0.9064
Epoch 2/10
44/44 [==============================] - 1s 11ms/step - loss: 0.0596 - accuracy: 0.9764
Epoch 3/10
44/44 [==============================] - 1s 18ms/step - loss: 0.0428 - accuracy: 0.9850
Epoch 4/10
44/44 [==============================] - 1s 16ms/step - loss: 0.0316 - accuracy: 0.9864
Epoch 5/10
44/44 [==============================] - 1s 15ms/step - loss: 0.0233 - accuracy: 0.9921
Epoch 6/10
44/44 [==============================] - 1s 18ms/step - loss: 0.0189 - accuracy: 0.9921
Epoch 7/10
44/44 [==============================] - 1s 16ms/step - loss: 0.0241 - accuracy: 0.9921
Epoch 8/10
44/44 [==============================] - 1s 14ms/step - loss: 0.0154 - accuracy: 0.9936
Epoch 9/10
44/44 [==============================] - 1s 15ms/step - loss: 0.0172 - accuracy: 0.9936
Epoch 10/10
44/44 [==============================] - 0s 10ms/step - loss: 0.0190 - accuracy: 0.9943
```

Fig 6. Training procedure

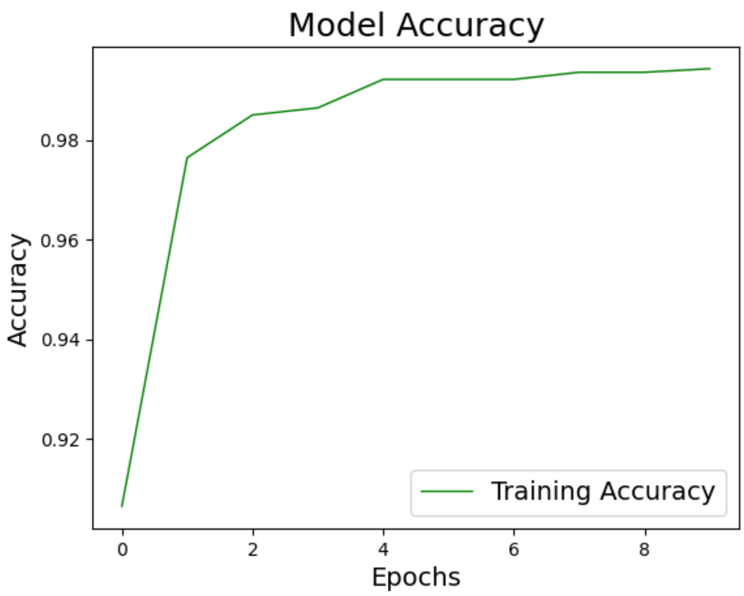


Fig 7. Model Accuracy

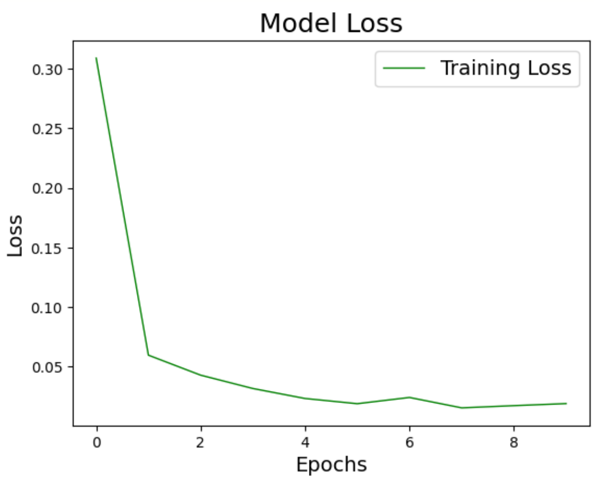


Fig 8. Model loss

```
44/44 [==============================] - 0s 2ms/step - loss: 0.0068 - accuracy: 0.9979
train_loss, train accuracy [0.006795268040150404, 0.9978571534156799]
```

Training precision and the loss are two frequent measures of a model's training performance. Definitions of each indicator are provided below.

1. Training Accuracy: The fraction of training instances that have been successfully categorised is represented by this measure. It is determined by dividing the number of accurate predictions by the total amount of training set instances. A better match between the training data and the model is indicated by a greater training accuracy.

2. Training Loss: The gap between the training data's anticipated and actual values (ground truth) is the focus of this metric's measurement. It indicates the degree to which the model's predictions correspond with the actual aims. Binary cross-entropy is a widely used loss function in classification problems. During training, success is measured by how near the model's predictions are to the actual values, or "loss."

We may learn about your model's learning and improvement over time by comparing accuracy and loss values collected during separate training epochs.

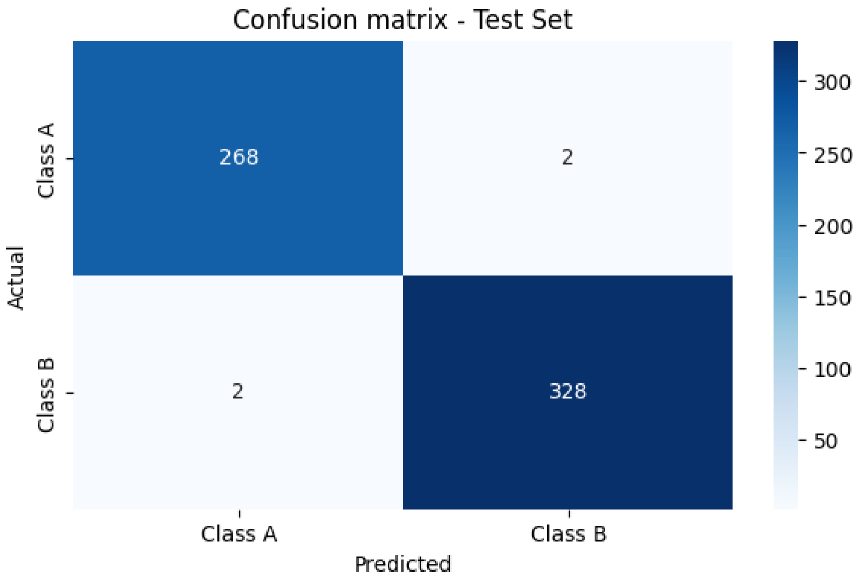


Fig 9. Confusion matrix of the proposed model

In classification tasks, the efficiency of a model may be evaluated with the use of a confusion matrix, which displays the percentage of cases properly categorized into each class. Dissecting the numbers in this matrix of confusion.

True Positive (TP): The proportion of class A occurrences that were accurately labeled as such. In this case, TP = 268.

False Positive (FP): The amount of cases that should have been placed in class B but were instead placed in class A. In this case, FP = 2.

True Negative (TN): The number of occurrences that should have been categorised as part of class B but were instead appropriately assigned to class B. In this case, TN = 328.

False Negative (FN): The amount of cases that should have been assigned to Class B but were instead assigned to Class A. In this case, FN = 2.

- Accuracy: The percentage of successful forecasts as a share of total forecasts.
- Precision: Number of true positives indicates the percentage of predictions that came true.
- Recall: Evaluation of the model's precision in detecting positive examples.
- F1-Score: a point when accuracy and recall are equal, achieving harmony.
- R2 Score: A measure of how much variation in a dependent variable can be explained by changes in other variables; sometimes called the coefficient of determination. In this case, R2 equals 0.9730.
- MAE (Mean Absolute Error): The mean absolute deviation from forecasted to observed values. The value of the MAE is 0.0066.

- MSE (Mean Squared Error): A measure of the typical squared disparity between expected and realized outcomes. In this case, the MSE is 0.0066.
- RMSE (Root Mean Squared Error): The average squared deviation between forecasted and observed values, expressed as a square root [24]. RMS Error = 0.0816.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\mathrm{yi} - \mathrm{y'i})^2 \quad (3)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\mathrm{yi} - \mathrm{y'i})^2} \quad (4)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\mathrm{yi} - \mathrm{y'i}| \quad (5)$$

```
              precision    recall  f1-score   support

         0.0       0.99      0.99      0.99       270
         1.0       0.99      0.99      0.99       330

    accuracy                           0.99       600
   macro avg       0.99      0.99      0.99       600
weighted avg       0.99      0.99      0.99       600
```

Fig 10. Validation parameters

Table I. Error details

Parameter	Value
R2 score	0.97
MAE	0.007
MSE	0.007
RMSE	0.081

Table II. Accuracy evaluation

Disease	**Accuracy(%)**
SVM	91.50
Random Forest	86.23
Naïve Bayes	89.26
FOA-SVM	95.47
Proposed ABC-CNN	**99.33**

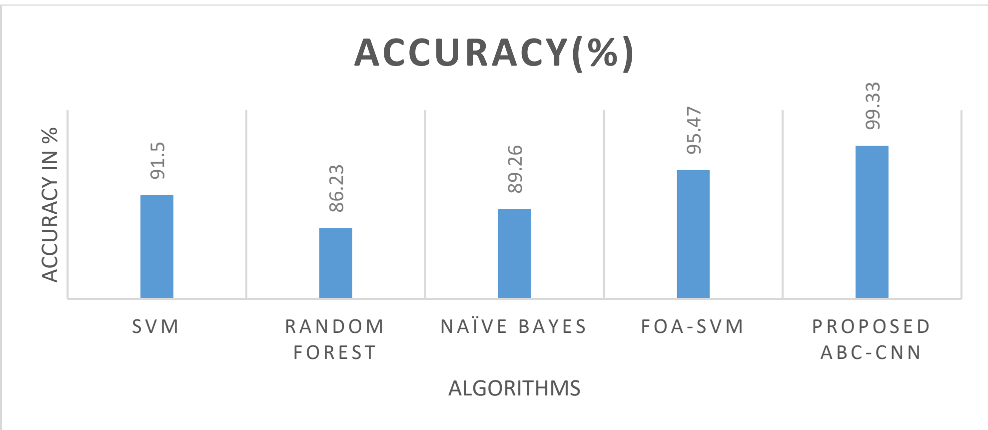


Fig 11. Accuracy tests of several algorithms

Table III provides a comparison of the recommended SDL to SVM, RF, & Nave Bayes. And as can be shown in Figure 11, the proposed strategy performs noticeably better than other approaches.

4. Conclusion

In rural India, agriculture is the main source of income, and the economy cannot function without it. About two-thirds of the workforce in the primary as well as secondary industries works in agriculture. As a consequence, a growing number of farmers are turning to innovative techniques like indoor farming to boost productivity. In this research, we use a dataset from kaggle to forecast the yield of tomatoes grown indoors using a hybrid approach: the ABC-CNN classification algorithm. The prediction method makes use of the Convolutional Neural Network (CNN) model and the Artificial Bee Colony (ABC) algorithm with the Adam optimizer. The results demonstrate that our suggested technique is more precise and efficient than competing methods. In the future, we want to experiment with different deep learning algorithms to see which ones provide the most accurate predictions of crop yields.

References

[1] Li, D., Xu, L., Tan, C., Goodman, E. D., Fu, D., & Xin, L. (2015). Digitization and visualization of greenhouse tomato plants in indoor environments. *Sensors (Switzerland)*, *15*(2), 4019–4051. https://doi.org/10.3390/s150204019

[2] Verma, S., Chug, A., & Singh, A. P. (2018). Prediction Models for Identification and Diagnosis of Tomato Plant Diseases. *2018 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2018*, *April 2019*, 1557–1563. https://doi.org/10.1109/ICACCI.2018.8554842

[3] Basavaiah, J., & Arlene Anthony, A. (2020). Tomato Leaf Disease Classification using Multiple Feature Extraction Techniques. *Wireless Personal Communications*, *115*(1), 633–651. https://doi.org/10.1007/s11277-020-07590-x

[4] Sagar, S., & Singh, J. (2023). An experimental study of tomato viral leaf diseases detection using machine learning classification techniques. *Bulletin of Electrical Engineering and Informatics*, *12*(1), 451–461. https://doi.org/10.11591/eei.v12i1.4385

[5] Liu, J., & Wang, X. (2020). Tomato Diseases and Pests Detection Based on Improved Yolo V3 Convolutional Neural Network. *Frontiers in Plant Science*, *11*(June), 1–12. https://doi.org/10.3389/fpls.2020.00898

[6] Chen, Y. M., Zu, X. P., & Li, D. (2020). Identification of Proteins of Tobacco Mosaic Virus by Using a Method of Feature Extraction. *Frontiers in Genetics*, *11*(October), 1–11. https://doi.org/10.3389/fgene.2020.569100

[7] Mokhtar, U., El-Bendary, N., Hassenian, A. E., Emary, E., Mahmoud, M. A., Hefny, H., Tolba, M. F., Mokhtar, U., Hassenian, A. E., Emary, E., & Mahmoud, M. A. (2015). SVM-Based detection of tomato leaves diseases. *Advances in Intelligent Systems and Computing*, *323*, 641–652. https://doi.org/10.1007/978-3-319-11310-4_55

[8] Appalanaidu, M. V., &Kumaravelan, G. (2021). Plant leaf disease detection and classification using machine learning approaches: a review. *Innovations in Computer Science and Engineering: Proceedings of 8th ICICSE*, 515-525.

[9] Agarwal, M., Gupta, S. K., & Biswas, K. K. (2020). Development of Efficient CNN model for Tomato crop disease identification. *Sustainable Computing: Informatics and Systems*, *28*, 100407. https://doi.org/10.1016/j.suscom.2020.100407

[10] Chowdhury, M. E. H., Rahman, T., Khandakar, A., Ayari, M. A., Khan, A. U., Khan, M. S., Al-Emadi, N., Reaz, M. B. I., Islam, M. T., & Ali, S. H. M. (2021). Automatic and Reliable Leaf Disease Detection Using Deep Learning Techniques. *AgriEngineering*, *3*(2), 294–312. https://doi.org/10.3390/agriengineering3020020

[11] Zhu, H., Cen, H., Zhang, C., & He, Y. (2016). Early detection and classification of tobacco leaves inoculated with tobacco mosaic virus based on hyperspectral imaging technique. *2016 American Society of Agricultural and Biological Engineers Annual International Meeting, ASABE 2016*, 1–7. https://doi.org/10.13031/aim.20162460422

[12] Shoaib, M., Hussain, T., Shah, B., Ullah, I., Shah, S. M., Ali, F., & Park, S. H. (2022). Deep learning-based segmentation and classification of leaf images for detection of tomato plant disease. *Frontiers in Plant Science*, *13*(October), 1–18. https://doi.org/10.3389/fpls.2022.1031748

[13] Sharma, P., Berwal, Y. P. S., &Ghai, W. (2020). Performance analysis of deep learning CNN models for disease detection in plants using image segmentation. *Information Processing in Agriculture*, *7*(4), 566–574. https://doi.org/10.1016/j.inpa.2019.11.001

[14] Gomaa, A. A., & El-Latif, Y. M. A. (2021). Early Prediction of Plant Diseases using CNN and GANs. *International Journal of Advanced Computer*

Science and Applications, *12*(5), 514–519. https://doi.org/10.14569/IJACSA.2021.0120563

[15] Li, L., Zhang, S., & Wang, B. (2021). Plant Disease Detection and Classification by Deep Learning - A Review. *IEEE Access*, *9*(Ccv), 56683–56698. https://doi.org/10.1109/ACCESS.2021.3069646

[16] Jayswal, H. S., &Chaudhari, J. P. (2020). Plant leaf disease detection and classification using conventional machine learning and deep learning. *International Journal on Emerging Technologies*, *11*(3), 1094-1102.

[17] Mishra, V., Jain, R., Chahande, M., & Dubey, A. K. (2021). Tomato Leaf Features Extraction for Early Disease Detection. In *Proceedings of International Conference on Machine Intelligence and Data Science Applications: MIDAS 2020* (pp. 255-265). Springer Singapore.

[18] Sun, J., Di, L., Sun, Z., Shen, Y., & Lai, Z. (2019). County-level soybean yield prediction using deep CNN-LSTM model. *Sensors (Switzerland)*, *19*(20), 1–21. https://doi.org/10.3390/s19204363

[19] Ramli, M. S., Zin, Z. M., Suleiman, R. F. R., Ibrahim, N., &Ngah, L. A. (2020). Review of machine learning implementation in advanced tomato plant disease detection and recognition. *Journal of Green Engineering*, *10*(12), 13032–13048.

[20] Kumar Mamindla, A. (2022). *a Study: Mammogram Image Segmentation and Classification Based on Abc Algorithm and Artificial Neural Networks*. *November*. https://www.researchgate.net/publication/365301410

[21] Xie, F., Li, F., Lei, C., Yang, J., & Zhang, Y. (2019). Unsupervised band selection based on artificial bee colony algorithm for hyperspectral image classification. *Applied Soft Computing Journal*, *75*, 428–440. https://doi.org/10.1016/j.asoc.2018.11.014

[22] Sen, S. Y., &Ozkurt, N. (2020). Convolutional Neural Network Hyperparameter Tuning with Adam Optimizer for ECG Classification. *Proceedings - 2020 Innovations in Intelligent Systems and Applications Conference, ASYU 2020*, *978*. https://doi.org/10.1109/ASYU50717.2020.9259896

[23] Khaki, S., Wang, L., &Archontoulis, S. V. (2020). A CNN-RNN Framework for Crop Yield Prediction. *Frontiers in Plant Science*, *10*(January), 1–14. https://doi.org/10.3389/fpls.2019.01750

[24] Oikonomidis, A., Catal, C., &Kassahun, A. (2022). Hybrid Deep Learning-based Models for Crop Yield Prediction. *Applied Artificial Intelligence*, *36*(1). https://doi.org/10.1080/08839514.2022.2031823

[25] Nevavuori, P., Narra, N., & Lipping, T. (2019). Crop yield prediction with deep convolutional neural networks. *Computers and Electronics in Agriculture*, *163*(April), 104859. https://doi.org/10.1016/j.compag.2019.104859

[26] Wang, Y., Xiao, Z., & Cao, G. (2022). A convolutional neural network method based on Adam optimizer with power-exponential learning rate for bearing fault diagnosis. *Journal of Vibroengineering*, *24*(4), 666–678. https://doi.org/10.21595/jve.2022.22271

[27] BANERJEE, S., BHARADWAJ, A., GUPTA, D., & PANCHAL, V. K. (2014). Remote Sensing Image Classification Using Artificial Bee Colony Algorithm. *International Journal of Computer Science and Informatics*, *3*(3), 155–160. https://doi.org/10.47893/ijcsi.2014.1137

[28] Muruganantham, P., Wibowo, S., Grandhi, S., Samrat, N. H., & Islam, N. (2022). A Systematic Literature Review on Crop Yield Prediction with Deep Learning and Remote Sensing. *Remote Sensing*, *14*(9). https://doi.org/10.3390/rs14091990