Plants Monitoring API to Detect Tomato Leaf Diseases using Deep-Learning Algorithms.

Ayman Moustafa, AbdulRahman Alsewari, Sara Hassan

College of Computing, Faculty of Computing, Engineering and the Built of Environment, Birmingham City University, Birmingham, United Kingdom
ayman.moustafa@mail.bcu.ac.uk, alsewari@gmail.com, sara.hassan@bcu.ac.uk

Abstract. In the presented study, the challenge of detecting tomato leaf diseases, crucial for sustainable agriculture, is addressed. A Convolutional Neural Network (CNN) model has been developed, demonstrating a high accuracy rate of 97.29% on a variety of test datasets. The methodology employed includes a comprehensive literature review, meticulous collection and augmentation of datasets, and the development of an advanced CNN model. Through these techniques, early and precise detection of diseases in tomato plants is facilitated. The contributions of this research are pivotal in transforming agricultural practices, as evidenced by enhanced crop health, increased yield, and promotion of sustainable farming methods. The significance of this work lies in its potential to contribute to global food security and the evolution of agriculture in the digital era.

Keywords: CNN, Machine learning, Deep learning, Tomato disease detection.

1 Introduction

Artificial Intelligence (AI) has been a game-changer in detecting plant diseases, offering solutions that are efficient, accurate, and sustainable [1]. Utilizing AI-based technologies, especially computer vision and machine learning, allows for the prompt and precise identification of plant diseases by analyzing images of leaves, stems, or whole plants. This approach significantly outperforms traditional methods like manual inspection or symptom-based diagnosis, which are generally slow and susceptible to human errors. AI plays an essential role in preserving the Earth's sustainability in several ways. Prompt disease detection through AI enables quick response, helping farmers to address affected plants or produce before the disease spreads extensively. This reduces crop losses, boosts yield, and supports food security, crucial in a world facing the challenge of feeding an ever-growing population. Moreover, AI-driven disease diagnosis minimizes excessive pesticide use, conserving resources, diminishing pollution, and fostering healthier ecosystems. Traditional methods often lead to the overuse of pesticides, negatively impacting beneficial insects and disrupting ecological balance.

Plants are indispensable to human existence. Since the dawn of time, humans have depended on cultivating plants for sustenance and survival. Although plant cultivation may seem straightforward, plants are susceptible to various diseases [2]. Food insecurity, a significant global challenge, is exacerbated by widespread plant diseases [3]. Numerous studies have focused on detecting and classifying plant diseases, aiming to protect the environment and secure our food sources. Identifying plant diseases is crucial for minimizing agricultural losses and ensuring global food safety [4]. Crop diseases pose a severe threat to global food security, necessitating immediate and accurate detection for effective control. Ubbens' research introduced a deep learning framework for plant phenotyping applications, such as disease detection, using deep convolutional neural networks (CNNs) to analyze plant images for precise disease identification [5]. However, the absence of necessary infrastructure in many regions hinders early disease diagnosis. This article aims to develop an image-based plant disease diagnosis system using advancements in deep learning. The system, designed to be user-friendly for farmers, leverages deep CNNs trained on diverse datasets for early disease detection and improved crop management. CNNs are engineered to learn spatial features of the targeted class or quantity, such as edges, corners, textures, or larger shapes [6].

Crop diseases significantly threaten global food security, causing major yield losses and affecting agricultural communities worldwide. However, the lack of readily available infrastructure for disease diagnosis often delays timely interventions, especially in areas with limited resources. The integration of smartphone technology and deep learning presents a promising solution. An image-based plant disease detection system, harnessing the ubiquity of smartphones and the capabilities of deep CNNs, can provide quick and precise diagnoses. This empowers farmers and facilitates effective disease control practices.

The selection of CNNs for this article is grounded in their unparalleled image processing capabilities, high accuracy and efficiency, adaptability to diverse disease manifestations, reduction in false diagnoses, and scalability for real-world applications. These attributes make CNNs the optimal choice for advancing the field of tomato leaf disease detection, ultimately contributing to more sustainable and productive agricultural practices.

2 Related Works

A primary driving force behind the investigation of alternative techniques is the difficulties posed by the conventional human method of diagnosing plant diseases, which primarily entails visual inspection. This traditional approach is frequently expensive, time-consuming, difficult, and prone to subjective opinions. Numerous machine learning techniques have been developed to improve accuracy, cost-effectiveness, and objectivity while mitigating these shortcomings. Research has indicated, for example, that Support Vector Machine (SVM) classifiers are capable of successfully distinguishing between rice seedlings with Bakanae disease and healthy plants. This approach has been found to be more effective than the conventional visual evaluation, providing quicker and more impartial results [7]. Another study, using three different classifiers (SVM, KNN, and probabilistic neural network) and five different types of diseases affecting soybean plants, was carried out by [8] to minimize the requirement for human intervention. The authors acknowledged the necessity of background reduction and segmentation while highlighting the significance of feature extraction. For the classification phase, the SVM technique with various kernel functions is employed. For both the training and testing phases, datasets of 800 healthy and infected tomato leaf images were employed. The presented approach's performance is evaluated using the N-fold cross-validation technique. The proposed classification strategy achieved a classification accuracy of 99.83% utilizing a linear kernel function, according to the experimental data [9].

Residual Network (ResNet) is a deep learning model used in computer vision. It's a Convolutional Neural Network (CNN) architecture with hundreds or thousands of convolutional layers. According to [10] , the ResNet34 network produced flawless outcomes with a 99.7% accuracy.
Baheti described a method for applying Machine Learning algorithms to predict and diagnose tomato plant leaf disease early [11]. They worked on a dataset and performed various operations on it, such as feature extraction, testing, and so on, using different 8 machine learning methods. The classification accuracy of tomato leaf disease was found to be between 67% and 99%. The Random Forest algorithm achieves the maximum accuracy of 98% for ten different diseases studied.

There are numerous established techniques for detecting plant diseases using machine learning. Some of the most popular methods are:

- Traditional machine learning algorithms: These algorithms include support vector machines (SVM), random forests, decision trees, and k-nearest neighbors. These algorithms have been used for many years to detect plant diseases and have been found to be useful in some circumstances. However, they can be sensitive to noise and outliers, and they may not be able to attain high accuracy for every kind of plant disease [12].
- Deep learning algorithms: Deep learning algorithms are a sort of machine learning that employs artificial neural networks to learn from data. Deep learning algorithms have been demonstrated to attain cutting-edge accuracy in plant disease detection. They do, however, require huge datasets for training and can be computationally expensive to train and deploy [13] .
- Transfer learning: Transfer learning is a technique in which a model trained on one task is utilized as a starting point for training a model on another activity. By applying a model that has been trained on a large dataset of healthy and sick plant images, transfer learning can be utilized to increase the accuracy of plant disease detection algorithms [14].
- Ensemble learning: Ensemble learning is a technique for improving accuracy by combining the predictions of multiple models. By pooling the predictions of different machine learning algorithms, ensemble learning can be used to increase the accuracy of plant disease detection models [15].

3 CNN Model Development

A detailed and informative flowchart illustrating the architecture of a Convolutional Neural Network (CNN) used for tomato leaf disease detection (see Fig. 1). This flowchart includes various elements such as the preprocessing and input process, the different layers of the CNN, data flow through the network, output interpretation, and its integration with the overall study's methodology.

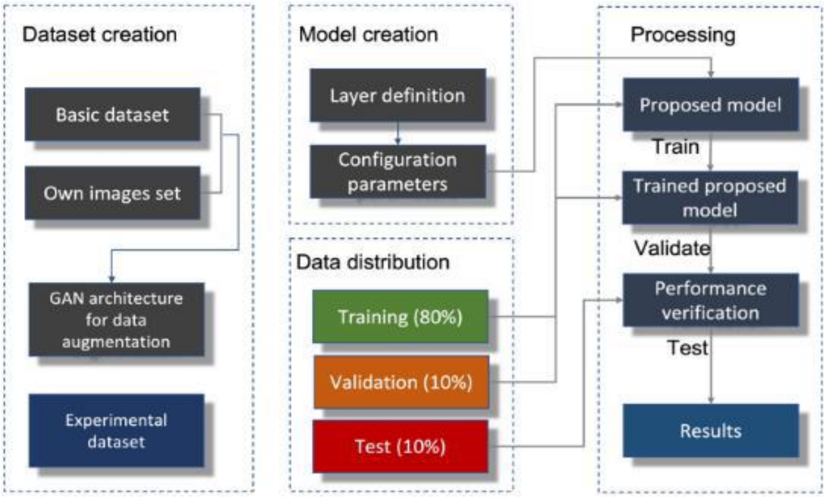

Fig. 1. Representation of the proposed architecture for tomato disease detection

3.1 Data Collection

The study takes significant advantage of the PlantVillage dataset, a large and diverse archive of images and related metadata representing a wide range of plant diseases, including those affecting tomato plants. The PlantVillage dataset is useful for training and evaluating the CNN-based disease detection model since it contains a wealth of real-world examples that reflect the complexity encountered by farmers.
This dataset is of size 690 MB and was obtained from Kaggle. It offers a comprehensive collection of high-resolution images depicting various plant leaves classified as healthy or diseased. The dataset, acquired on [01/06/2023], serves as a valuable resource for training and evaluating our CNN models, enabling us to develop accurate and efficient methods for plant disease diagnosis and monitoring. The dataset includes a diverse range of plant species, including significant crops such as tomatoes, potatoes, and maize. It includes images of both the leaves and the entire plant. We've selected the images related to healthy tomato leaves and those related to two main types of tomato leaf diseases which are "Early Blight" and "Late Blight". The total number of images used in the model is 4500 images. The dataset's geographic and climate diversity is useful for training machine learning models to identify a wide range of tomato diseases and environmental conditions. It enables a more comprehensive approach to treating

plant health issues facing farmers globally. Figure 2 shows sample images from the dataset.

Fig. 2. Dataset Samples
Dataset link: https://www.kaggle.com/datasets/emmarex/plantdisease

Data Pre-processing

Data pre-processing is the critical phase for improving raw data for analysis. Cleaning (fixing missing values, deleting duplicates, and dealing with outliers), transforming (turning categories to numbers, scaling features), and reducing data (dimensionality, sampling) are all part of the process. It attempts to increase data quality and analytical availability resulting in more accurate research outcomes. The collected dataset undergoes preprocessing to prepare it for training and evaluation. This step involves standardizing the image sizes, converting them to a suitable format, and ensuring consistency in image quality and resolution.

BATCH_SIZE = 32

The batch size is the number of training examples utilized in a single training iteration. In our case, the model will process 32 images at a time before modifying the weights according to the gradients calculated. Batch training facilitates the use of parallelism in modern hardware while also introducing a form of regularization by adding noise to the updates.

IMAGE_SIZE = 256x256

The dimensions to which all input images have been compressed before they are introduced into the model are referred to as image size. The images are resized to 256x256 pixels in this model. It is critical to resize the images to a constant size for consistent network processing.

CHANNELS=3

The present researcher is using the RGB images which has 3 color channels which are :Red ,Green, and Blue .

Epochs=50

An epoch is one whole cycle across the entire training dataset. Training a model entails iterating through the dataset for a set number of epochs. We are training the model for

50 epochs in this scenario, which means the model will see the complete training dataset 50 times throughout training. Additionally, the dataset is divided into three subsets :
• training set for model development (80% of the dataset was used for training which is about 3584 images)
• validation set for model performance evaluation and hyper parameter tuning (448 images were used for validation)
• testing set for final model assessment (448 images were used for the testing purposes).

3.2 Model Architecture Selection

Deep Learning
Deep Learning is a branch of machine learning which excels at analyzing patterns based on a lot of data. Using an artificial neural network with three or more layers, which extracts many visual properties from each layer, objects can be recognized from photos [16].
An appropriate deep convolutional neural network (CNN) architecture is selected for tomato leaf disease classification. The performance of commonly used architectures such as VGG, ResNet, Inception, and EfficientNet in image classification tasks are also taken into consideration. The architecture chosen should find a compromise between accuracy and computing efficiency.
Date Augmentation

```
data_augmentation = tf.keras.Sequential([
layers.experimental.preprocessing.RandomFlip("horizontal_and_vertical"),
  layers.experimental.preprocessing.RandomRotation(0.2),])
```

An important feature of this research is the effect of data augmentation on the effectiveness of deep-learning algorithms for plant disease diagnosis. Data augmentation is a technique for artificially expanding the training dataset by transforming the original images with rotations, flips, and scaling. The major goal is to determine how these supplemented data variances affect the ability of the model to detect and classify plant diseases. We may determine the value of data augmentation in improving the model's generalization, robustness, and overall performance in real-world disease detection scenarios by systematically evaluating the impact it has. This study sheds light on the important role of data augmentation in boosting the accuracy and reliability of deep learning models for plant disease evaluation, which is critical for sustainable agriculture and food security.
This data augmentation pipeline represented by the code above is used as a preprocessing step for training data. When running training images through this pipeline during the data loading process, it will add these random flips and rotations to the images, which can assist enhancing the generalization of the deep learning model by exposing it to a wider variety of variables in the training data.
By combining these processes, we are performing on-the-fly data augmentation on the training dataset during the process of loop over it, which helps the variety of the training

data and improve the robustness of the deep learning model. Furthermore, by overlapping data loading and model calculation, prefetching the data batches can help reduce training time.

3.3 Model Evaluation and Visualization

In our CNN developed model, we performed model evaluation and visualization after training a convolutional neural network (CNN) for an image classification task. Here's what's considered in the model evaluation and visualization:

Training and Validation Loss/Accuracy

Throughout training, we track the accuracy and loss for training and validation over a number of epochs. The model's performance in learning from the training data and overfitting (high training accuracy but low validation accuracy) are both evaluated using the information presented.

Test Set Evaluation

Subsequent to the training phase, the model's efficacy was assessed utilizing the 'evaluate' method on an independent test dataset (test_ds). This evaluation yields critical metrics such as loss and accuracy, pertinent to data previously unexposed to the model, thereby furnishing insights into the model's applicability and predictive robustness on novel, unexplored datasets.

Test Set Visualization

- Visually examining the model's predictions on a sample from the test dataset.
- Display the actual image for each of the images in the sample.
- Display the anticipated class label along with its probability (confidence) score.
- Show the actual class name for the purpose of comparison.

Confidence Score

A confidence score is a numerical indicator of a machine learning model's level of confidence in its prediction. High scores suggest a high degree of confidence, whilst low levels denote a degree of uncertainty. For each prediction, we determine and present the confidence score as a percentage.

Visualizing Predictions

We visualize a grid of images from the test dataset as presented in Figure 3, which shows the actual and predicted class labels, along with confidence scores. This enables us to evaluate the model's effectiveness on specific images on a qualitative level.

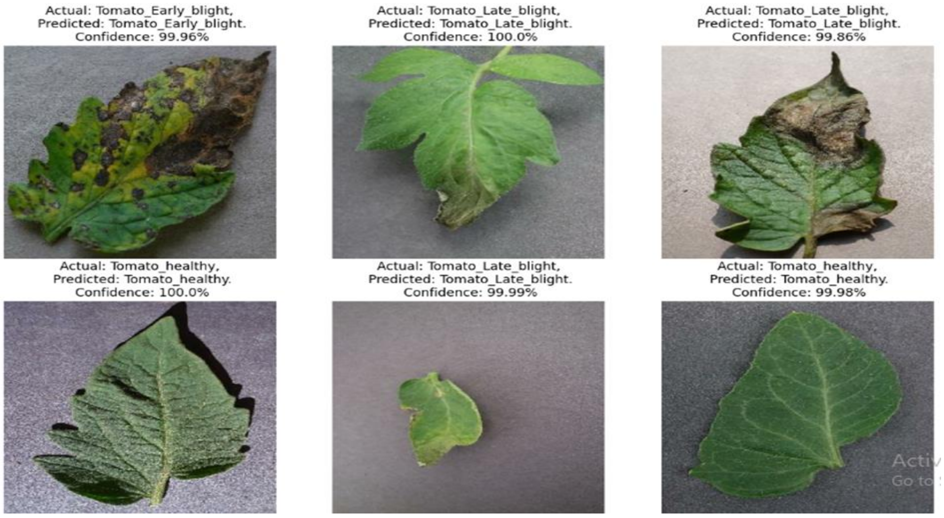

Fig. 3. Images showing the actual and predicted class labels, with confidence scores.

As shown in Fig. 3, the model has showed an accurate prediction for a grid of images from the test dataset with high confidence scores.

4 Deployment Strategy

Deployment Environment Optimization

Our CNN model, originally developed in Python within Jupyter notebooks, has been transitioned to a more robust deployment framework. Utilizing FastAPI and Uvicorn as the ASGI server, we've created a seamless environment for effective HTTP request handling and deployment. This setup enhances the model's responsiveness and reliability in a production setting.

Advanced API Integration

The integration of FastAPI endpoints marks a significant improvement in our model's interactivity. These endpoints were rigorously tested with Postman, ensuring robustness and reliability in API communication. This careful testing phase guarantees smooth and efficient model interactions in real-world applications.

User Interface Development

A sophisticated user interface, developed using React.js, offers an intuitive experience for users. This interface simplifies the process of uploading images for analysis and obtaining predictions from the model. The tool, showcased in Fig. 4, serves as the primary point of interaction with the deployed model, emphasizing user-friendliness and efficiency.

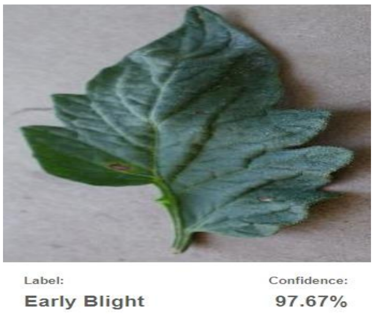

Fig. 4. Images showing the actual and predicted class labels, with confidence scores.

5 Evaluation and Testing

Data Segmentation Strategy

Dataset Splitting: We divided the dataset into training (80%), validation (10%), and testing (10%) segments. This distribution ensures comprehensive training on a diverse dataset while maintaining unbiased evaluation on unseen data.

Rigorous Model Training Process

Training with TensorFlow: Utilizing TensorFlow, we implemented a rigorous training regimen. This included selecting specific optimizers, loss functions, and evaluation metrics tailored to our objectives.

Iterative Learning through Epochs: The model underwent a 50-epoch training phase, enabling iterative learning and refinement based on the training data.

Detailed Model Evaluation

Performance Assessment: The model's effectiveness was thoroughly evaluated on the test dataset, providing a clear measure of its real-world applicability.

Visualization of Progress: Training and validation curves were plotted to depict the model's learning trajectory and overall performance dynamics.

Insights from Performance Metrics

Learning Curve Analysis: The model showed substantial progress during training, improving from an initial accuracy of 66.29% to 98.21% on the validation set. This improvement underscores its capability in precise disease classification.

Monitoring Loss Trends: We closely observed the patterns of training and validation loss. The consistent decrease in training loss and stable validation loss indicates successful pattern recognition and the avoidance of overfitting.

Validation through Independent Testing

Testing on Separate Data: The model underwent further validation using a distinct test dataset, where it achieved an accuracy rate of 97.29%. This result confirms the model's robustness and generalizability.

Visualization of Training Process

Graphical Representations: To visually articulate the training journey, we created graphs that illustrate training and validation accuracy, alongside training and validation loss (refer to Fig. 5).

The model developed in this study for detecting tomato leaf disease has demonstrated impressive performance, underscoring its significant potential to enhance disease control methods in tomato farming.

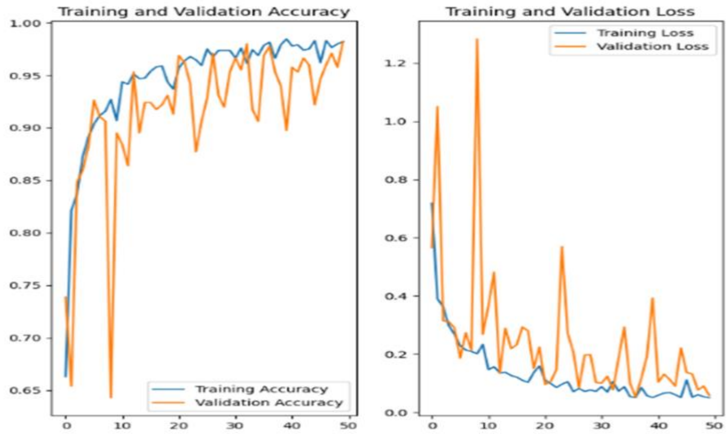

Fig. 5. Graphs showing training and validation accuracy as well as the training and validation loss.

6 Results Discussion

The model developed in our study for detecting tomato leaf disease has exhibited notable success, emphasizing its potential to revolutionize disease control methods in tomato farming. This section delves into the detailed outcomes of the model's testing and evaluation.

Model Performance

Efficient Disease Classification: Trained over 50 epochs with batches of 32, our Convolutional Neural Network (CNN) model demonstrated high proficiency in disease classification. It achieved an impressive accuracy rate of approximately 97.29% on the test dataset, underscoring its effective learning capabilities.

Progressive Loss Reduction: The loss curve of the model consistently trended downwards, showcasing its adaptability in reducing classification errors progressively.

Accuracy and Reliability

Impact on Agriculture: The model's remarkable accuracy has profound implications for practical agricultural scenarios. By enabling early and accurate detection of diseases in tomato plants, farmers can significantly enhance crop yields. The model's precision in minimizing both false positives and negatives is crucial for ensuring effective disease management and reducing unnecessary treatments.

Generalization Capabilities

Overfitting Mitigation: The model's exceptional ability to generalize to new, unseen data significantly alleviates concerns about overfitting. Its consistent performance across diverse tomato leaf image datasets makes it a valuable tool for real-world agricultural applications.

Practical Applications

Early Disease Detection in Agriculture: The model offers a range of practical applications in the agricultural sector. Farmers can leverage this technology to detect diseases at an early stage, enabling timely interventions that improve crop health and yield.

Enhancing Sustainable Farming Practices: The adoption of this model can contribute to more sustainable farming practices. By reducing the reliance on pesticides and promoting efficient disease control, it aligns with the goals of sustainable agriculture, thereby supporting environmental health and crop productivity.

7 Limitations and Challenges

While the model has shown remarkable success, it is critical to recognize and address various threats to its validity, which include both internal and external factors. These threats may have an impact on the model's precision and generality and mitigating them is critical for further improvement. These threats can be classified into two categories:

Internal Threats to Validity

Data limitations: One notable internal threat is the availability and quality of the training dataset. Extending the dataset and ensuring its representativeness can help to improve the precision of the model.

Computational Resources: Access to more powerful computing resources can improve the model's performance even further, allowing for more extensive training and fine-tuning processes.

Hyperparameter Optimization: Another internal challenge is to investigate techniques for optimizing hyperparameters. Finding the best configuration for the model's parameters can have a big impact on the model's overall performance. In our study, the selection of hyperparameters for the CNN model, including the learning rate, batch size, and number of epochs, was based on a combination of empirical testing and best practices from existing literature, aiming to balance the model's learning efficiency with its ability to generalize effectively to new data.

This article addresses the issue of underfitting, where the model's simplicity might prevent it from capturing the complexities of the dataset, and outline our strategies for enhancing model complexity and training duration to ensure robust and accurate performance

External Threats to Validity

Generalizability: Ensuring the model's generalizability across different domains and contexts poses a danger from without. Its performance in various contexts needs to be evaluated to see how resilient it is.

Deployment Challenges: Making the move from theoretical deployment to practical deployment can provide unforeseen external obstacles. Scalability, real-time processing, and system integration problems are a few examples of these.

8 Conclusion

This study focuses on the huge agricultural challenge of tomato leaf disease detection, which is motivated by the need for novel solutions and the promise of machine learning in disease management. The researchers set out to create a powerful Convolutional Neural Network (CNN) model and accompanying tools for early and precise disease diagnosis. A literature analysis, dataset preparation, and CNN architecture construction were all part of the technique, which resulted in a user-friendly web-based application built with FastAPI and Uvicorn. On a different test dataset, the deployed CNN model demonstrated an impressive accuracy of roughly 97.29%, providing significant benefits for agriculture by enabling early disease identification and management. The study contributes to agriculture by using cutting-edge machine learning algorithms, improving crop management, and supporting sustainable practices by reducing pesticide use. Despite the advances, there are still constraints, such as the possibility of further improvement with a larger dataset and additional computational power. Enhancements include fine-tuning hyperparameters and increasing the dataset with different photos. The study suggests future research possibilities, such as expanding the model to include other tomato leaf diseases, improving performance through parameter modifications, and developing a user-friendly smartphone application for farmers to use in the field. Finally, this study demonstrates the revolutionary potential of machine learning in agriculture by addressing real-world problems and offering farmers with practical answers. As agriculture changes in the digital era, the study emphasises the significance of technology in maintaining global food security and sustainability.

References

1. Sharma, S., Verma, K. and Hardaha, P., 2023. Implementation of artificial intelligence in agriculture. Journal of Computational and Cognitive Engineering, 2(2), pp.155-162.
2. Chowdhury, M.E., Rahman, T., Khandakar, A., Ayari, M.A., Khan, A.U., Khan, M.S., Al-Emadi, N., Reaz, M.B.I., Islam, M.T. and Ali, S.H.M., 2021. Automatic and reliable leaf disease detection using deep learning techniques. AgriEngineering, 3(2), pp.294-312.
3. Ubbens, J.R. and Stavness, I., 2017. Deep plant phenomics: a deep learning platform for complex plant phenotyping tasks. Frontiers in plant science, 8, p.1190.
4. Mohanty, S.P., Hughes, D.P. and Salathé, M., 2016. Using deep learning for image-based plant disease detection. Frontiers in plant science, 7, p.1419.
5. Fuentes, A., Yoon, S., Kim, S.C. and Park, D.S., 2017. A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. Sensors, 17(9), p.2022.
6. Kattenborn, T., Leitloff, J., Schiefer, F. and Hinz, S., 2021. Review on Convolutional Neural Networks (CNN) in vegetation remote sensing. ISPRS journal of photogrammetry and remote sensing, 173, pp.24-49.
7. Chung, C.L., Huang, K.J., Chen, S.Y., Lai, M.H., Chen, Y.C. and Kuo, Y.F., 2016. Detecting Bakanae disease in rice seedlings by machine vision. Computers and electronics in agriculture, 121, pp.404-411.
8. Shrivastava, S., Singh, S.K. and Hooda, D.S., 2017. Soybean plant foliar disease detection using image retrieval approaches. Multimedia Tools and Applications, 76, pp.26647-26674.

9. Mokhtar, Usama & El-Bendary, Nashwa & Hassenian, AboulElla & Emary, Eid & Mahmood, Mahmood & Hefny, Hesham & Tolba, Mohamed. (2015). SVM-Based Detection of Tomato Leaves Diseases. In Intelligent Systems' 2014: Proceedings of the 7th IEEE International Conference Intelligent Systems IS'2014, September 24-26, 2014, Warsaw, Poland, Volume 2: Tools, Architectures, Systems, Applications (pp. 641-652). Springer International Publishing.
10. Tan, L., Lu, J. and Jiang, H., 2021. Tomato leaf diseases classification based on leaf images: a comparison between classical machine learning and deep learning methods. AgriEngineering, 3(3), pp.542-558.
11. Baheti, H., Thakare, A., Bhople, Y., Darekar, S. and Dodmani, O., 2022, April. Machine learning algorithm for detection and classification of tomato plant leaf disease. In 2022 IEEE 7th International conference for Convergence in Technology (I2CT) (pp. 1-7). IEEE.
12. Behmann, J., Mahlein, A.K., Rumpf, T., Römer, C. and Plümer, L., 2015. A review of advanced machine learning methods for the detection of biotic stress in precision crop protection. Precision agriculture, 16, pp.239-260.
13. Hinton, G., Vinyals, O. and Dean, J., 2015. Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531.
14. Maria, S.K., Taki, S.S., Mia, M.J., Biswas, A.A., Majumder, A. and Hasan, F., 2022. Cauliflower disease recognition using machine learning and transfer learning. In Smart Systems: Innovations in Computing: Proceedings of SSIC 2021 (pp. 359-375). Springer Singapore.
15. Kannan, E., 2022. An Efficient Deep Neural Network for Disease Detection in Rice Plant Using XGBOOST Ensemble Learning Framework. International Journal of Intelligent Systems and Applications in Engineering, 10(3), pp.116-128.
16. Hasan, M., Tanawala, B. and Patel, K.J., 2019, March. Deep learning precision farming: Tomato leaf disease detection by transfer learning. In Proceedings of 2nd international conference on advanced computing and software engineering (ICACSE).