International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Machine Learning Models for Plant Disease Detection and Classification

Nikhil S. Band[1], and Dr.Hare Ram Shah[2]

Submitted: 25/01/2024 **Revised**: 03/03/2024 **Accepted**: 11/03/2024

Abstract: The rise in the need for sustainable agricultural approaches has sparked a surge of curiosity in devising effective strategies for early identification and categorization of plant ailments. Machine learning (ML) methodologies have surfaced as potent instruments in this realm, thanks to their capacity to scrutinize extensive datasets and derive significant insights. This analysis offers a glimpse into the latest progressions in ML frameworks tailored for pinpointing and categorizing plant diseases. The exploration commences by delving into the significance of timely disease detection in agriculture and the hurdles linked with conventional techniques. It then outlines the basic concepts of ML and its applications in plant disease detection, emphasizing the role of feature extraction, feature selection, and classification algorithms. Several types of ML models commonly used in plant disease detection are examined, including supervised learning algorithms such as support vector machines (SVM), decision trees, random forests, and deep learning techniques like convolutional neural networks (CNNs) and recurrent neural networks (RNNs). The strengths and limitations of each approach are discussed, along with examples of their application in real-world scenarios. Furthermore, the review highlights the importance of dataset quality, size, and diversity in training ML models for plant disease detection. It also addresses challenges such as data imbalance, transfer learning, and model interpretability. In conclusion, this review provides insights into the current state-of-the-art ML techniques for plant disease detection and classification, along with future research directions aimed at improving model accuracy, robustness, and scalability. By leveraging the power of ML, stakeholders in agriculture can enhance crop yield, reduce economic losses, and promote sustainable farming practices.

Keywords: *Classification algorithms, Feature extraction, Machine learning, Supervised learning, Plant disease detection*

1. Introduction

In recent years, the agricultural sector has been increasingly challenged by the need for efficient and sustainable methods to combat plant diseases. Swift identification and precise categorization of these ailments stand as pivotal measures for curtailing crop losses, upholding food security, and advancing sustainable farming methodologies. Conventional disease diagnosis methods frequently hinge on visual examination by seasoned agronomists, a process susceptible to being time-intensive, subjective, and susceptible to human fallibility. As a result, there has been a growing interest in leveraging machine learning (ML) techniques to automate and enhance the process of plant disease detection and classification.

Machine learning, a facet of artificial intelligence, encompasses crafting algorithms that empower computers to glean insights and formulate forecasts or judgments from data without necessitating explicit programming. ML models have shown great promise in various domains, including computer vision, where they can analyze images and extract meaningful patterns. When it comes to identifying plant diseases, ML models can undergo training on expansive collections of images showcasing both healthy and diseased plants. Through this process, they grasp the unique characteristics that set apart various ailments.

The aim of this introduction is to offer a comprehensive glimpse into the involvement of machine learning models in the detection and classification of plant diseases. It will discuss the limitations of traditional methods, the potential benefits of ML-based approaches, and the challenges that need to be addressed in deploying such models in real-world agricultural settings.

Firstly, we will explore the shortcomings of conventional disease diagnosis techniques and the need for more efficient and accurate alternatives. Following this, we'll explore the fundamental tenets of machine learning and its integration within the realm of plant pathology. We'll examine a spectrum of ML models often employed for disease identification, exploring supervised learning techniques such as support vector machines and decision trees, alongside an exploration of advanced deep learning methodologies like convolutional neural networks.

Furthermore, we will highlight the importance of high-quality and diverse datasets in training ML models for plant disease detection. We will also address challenges such as data imbalance, transfer learning, and model interpretability, which are critical considerations in deploying ML-based solutions in real-world agricultural settings.

[1] *Assistant Professor ,Department of Information Technology, Prof Ram Meghe Institute of Technology and Research, Badnera, Amravati, Maharashtra, India 444701*
[2] *Professor, Department of Computer Science and Engineering, Sage University, Indore, India*
** Corresponding Author Email: nikhilband005@gmail.com*

In sum, this introduction lays the groundwork for a thorough exploration of how machine learning models are reshaping the landscape of plant disease identification and categorization. Through harnessing the capabilities of ML stakeholders in agriculture can potentially enhance crop yield, reduce economic losses, and promote sustainable farming practices in the face of emerging disease threats.=

2. Related Work

Plant disease detection encompasses the task of identifying and diagnosing illnesses that impact plants. Swift detection plays a pivotal role in curbing the dissemination of diseases and mitigating the extent of crop losses. One of the most common methods of detecting plant diseases is through visual inspection. Farmers, agronomists, and plant pathologists examine plants for symptoms such as discoloration, wilting, lesions, abnormal growth, and other indicators of disease. Plant diseases often exhibit specific symptoms that can help identify the type of disease present. Symptoms may include spots on leaves, stunted growth, yellowing of leaves, rotting, and more. Plant pathologists and researchers study these symptoms to identify the underlying causes.

In some cases, samples of plants showing symptoms of disease may be collected and sent to a laboratory for further analysis. Techniques such as microscopy, culturing pathogens, PCR (Polymerase Chain Reaction), ELISA (Enzyme-Linked Immunosorbent Assay), and DNA sequencing may be used to identify pathogens or other factors causing the disease. Remote sensing technologies, including satellite imagery and drones equipped with sensors, can be used to detect anomalies in plant health over large agricultural areas. These technologies can detect subtle changes in plant reflectance patterns, which may indicate the presence of disease or stress. Machine learning algorithms and artificial intelligence techniques are increasingly being used for automated plant disease detection. Such systems undergo training using extensive datasets comprising images portraying both healthy and afflicted plants. This training equips them with the capability to discern patterns and symptoms linked to a range of diseases. Once trained, these models can analyze images of plants and provide rapid diagnosis.

Diverse sensors can be implemented across agricultural fields to oversee environmental factors like temperature, humidity, soil moisture, and indicators of plant health. Changes in these parameters may signal the presence of disease or stress in plants. Integrated Pest Management (IPM) strategies combine multiple approaches to manage plant diseases, including cultural practices, biological control, chemical control (such as fungicides), and resistant plant varieties. Disease detection is a key component of IPM, enabling farmers to implement appropriate control measures in a timely manner. Overall, effective plant disease detection relies on a combination of traditional observation methods, advanced laboratory techniques, remote sensing technologies, and innovative approaches such as machine learning to accurately identify and manage plant diseases.

Machine learning (ML) techniques are increasingly being employed for identifying plant diseases owing to their adeptness in scrutinizing extensive image datasets and accurately classify plant health conditions. The first step in applying machine learning for plant disease detection is to collect a large dataset of images representing various plant diseases, as well as images of healthy plants. These images can be captured using cameras, smartphones, drones, or alternative imaging apparatus. Once the image dataset is collected, preprocessing steps may be applied to standardize image sizes, adjust lighting conditions, and remove noise or irrelevant background information. This aids in maintaining consistency and enhancing the efficacy of machine learning algorithms.

Machine learning algorithms require features to make predictions. In the context of plant disease detection, features can include colour histograms, texture descriptors, shape characteristics, and more. Techniques for feature extraction are employed to convert raw image data into a format compatible with machine learning algorithms. Plant disease detection can leverage a variety of machine learning models, encompassing CNNs, SVMs, decision trees, and ensemble methods. Notably, CNNs have exhibited outstanding efficacy in image classification endeavours, attributed to their capacity to directly acquire hierarchical features from image data. These models are trained on the pre-processed image dataset, with labelled images indicating whether each plant is healthy or diseased.

After training the machine learning model, it undergoes evaluation using a distinct validation dataset to gauge its effectiveness and fine-tune parameters if necessary. Once the model performs well on a separate test dataset, it's ready for real-world use. This could involve integrating it into software, apps, or systems that analyse plant images and give feedback to farmers.

Machine learning models for plant disease detection can be continuously improved by periodically updating the dataset with new images and retraining the model to adapt to emerging disease patterns or environmental conditions. By leveraging machine learning techniques, researchers and agricultural practitioners can develop efficient and accurate systems for detecting plant diseases, enabling early intervention and effective management strategies to safeguard crop health and optimize agricultural productivity.

Table 1. A comparative analysis of commonly employed machine learning methods for plant disease detection

Machine Learning Method	*Description*	*Advantages*	*Disadvantages*	*Application Examples*
Supervised Learning (e.g., SVM, Random Forests)	Trained on labeled data to classify plant images into diseased or healthy categories.	Effective for classification tasks. Relatively simple to implement.	Limited performance with complex image datasets. May require extensive feature engineering.	Classification of leaf images into multiple disease classes.
Convolutional Neural Networks (CNNs)	Deep learning architectures engineered to autonomously acquire hierarchical features from image datasets.	High accuracy in image classification tasks. Capable of discerning intricate patterns and correlations within datasets	Requires large amounts of labeled data for training. Computationally intensive and requires powerful hardware.	Detection of diseases in plants from leaf images with high accuracy.
Transfer Learning	Technique where pre-trained models are fine-tuned on plant disease datasets to leverage learned features.	Reduces the need for large labeled datasets. Speeds up model training process.	Limited to domains similar to pre-trained datasets. May require careful selection of pre-trained models and datasets.	Adapting models trained on generic image datasets (e.g., ImageNet) for plant disease detection tasks.
Unsupervised Learning (e.g., Clustering)	Identifies patterns and structures within unlabeled data to group similar instances together.	Does not require labeled data for training. Able to reveal concealed patterns within datasets	Limited applicability in plant disease detection without labeled data for validation. May produce ambiguous clusters without clear interpretation.	Identifying clusters of similar plant symptoms for further investigation.
Ensemble Methods	Integrates various machine learning models to enhance overall predictive accuracy.	Reduces overfitting and variance. Can capture diverse patterns and relationships in data.	Increased computational complexity and training time. Requires tuning of ensemble parameters.	Combining predictions from multiple classifiers to enhance disease detection performance.
Supervised Learning (e.g., SVM, Random Forests)	Trained on labeled data to classify plant images into diseased or healthy categories.	Effective for classification tasks. Relatively simple to implement.	Limited performance with complex image datasets. May require extensive feature engineering.	Classification of leaf images into multiple disease classes.

Plant diseases present substantial challenges to worldwide food security, highlighting the urgency for effective detection methods [Savary et al., 2019]. Machine learning offers promising solutions for automated and accurate disease detection in plants [Kamilaris and Prenafeta-Boldú, 2018]. Traditional methods such as visual inspection and laboratory tests suffer from subjectivity and time constraints [Haug et al., 2013]. Machine learning techniques have shown potential in overcoming these limitations by automating detection processes [Mohanty et al., 2016]. Plant disease classification has extensively utilized supervised learning algorithms like Support Vector Machines (SVM) and Random Forests [Fuentes et al., 2017]. Deep learning methodologies, notably CNNs, have showcased exceptional efficacy in image-based disease detection tasks [Liu et al., 2017].

Datasets like PlantVillage and FungiDB provide valuable resources for training and evaluating machine learning models in plant disease detection [Hughes and Salathe, 2015]. Data preprocessing techniques such as image augmentation and normalization are essential for improving model robustness and generalization [Barbedo, 2019]. Obstacles in utilizing machine learning for plant disease detection encompass the scarcity of adequately labeled datasets and environmental variability [Mwebaze et al., 2016]. Transferability of models across different plant species and diseases remains a significant limitation in real-world applications [Fuentes et al., 2017].

Recent research endeavours have delved into inventive techniques like transfer learning and ensemble methods to enhance the accuracy of disease detection [Ghosal et al., 2018]. Case studies demonstrate successful implementations of machine learning systems for early disease diagnosis and precision agriculture [Barbedo, 2019]. Assessment metrics such as accuracy, precision, recall, and F1-score serve as common yardsticks for gauging the effectiveness of machine learning models in disease detection tasks [Islam et al., 2020]. Benchmarking studies facilitate comparative analyses and help identify the most effective algorithms and methodologies [Kaur and Kaur, 2020].

Future research should focus on addressing data scarcity issues, improving model transferability, and integrating machine learning systems into agricultural practices [Kamilaris and Prenafeta-Boldú, 2018]. Machine learning holds great promise for revolutionizing plant disease detection and enhancing global food security in the coming years [Savary et al., 2019]. The comparison table 1 highlights various machine learning methods commonly used for plant disease detection.

3. Machine Learning Frameworks for Plant Disease Detection and Classification

Several machine learning frameworks and libraries are commonly used for plant disease detection and classification tasks. These frameworks provide a range of tools, algorithms, and APIs for building, training, and implementing machine learning models. TensorFlow stands as an open-source machine learning framework crafted by Google. It provides a robust ecosystem for constructing and implementing machine learning models, encompassing deep learning models for image classification tasks like plant disease detection. TensorFlow provides high-level APIs like Keras, streamlining the task of constructing and training deep learning models.

Table 2. Comparison of machine learning frameworks commonly used in plant disease detection and classification tasks

Feature / Framework	*TensorFlow*	*PyTorch*	*Scikit-learn*	*Keras*	*MXNet*	*Caffe*
Ease of Use	Moderate	Moderate	Easy	Easy	Moderate	Moderate
Flexibility	High	High	Moderate	High	High	Moderate
Deep Learning Support	Yes	Yes	Limited	Yes	Yes	Yes
Classical ML Algorithms	Limited	Limited	Yes	Limited	Limited	Limited
Community Support	High	High	High	High	Moderate	Moderate
Model Deployment	Yes	Yes	Yes	Yes	Yes	Yes
Performance	High	High	Moderate	Moderate	High	High

PyTorch emerges as an open-source machine learning library originating from Facebook's AI Research lab. It is acclaimed for its dynamic computational graph feature, which allows for more flexibility during model training. PyTorch provides modules and utilities for building convolutional neural networks (CNNs) and other deep learning architectures commonly used in plant disease classification tasks. Scikit-learn is a popular machine learning library in Python for classical machine learning algorithms. It furnishes straightforward and effective utilities for data preprocessing, feature extraction, and model training. While Scikit-learn may not be as focused on deep learning as TensorFlow or PyTorch, it offers various supervised and unsupervised learning algorithms suitable for plant disease classification tasks.

Keras represents a high-level neural networks API scripted in Python, adaptable for execution atop TensorFlow, Microsoft Cognitive Toolkit (CNTK), or Theano. It provides an easy-to-use interface for building deep learning models, including CNNs, recurrent neural networks (RNNs), and more. Keras is often used in conjunction with TensorFlow for rapid prototyping and experimentation with deep learning models. MXNet is an open-source deep learning framework nurtured by the Apache Software Foundation. It provides effective and scalable resources for constructing and implementing deep learning models across diverse platforms, including CPUs, GPUs, and cloud environments. MXNet supports multiple programming languages, including Python, Scala, and Julia, making it accessible to a wide range of users.

Caffe stands as a deep learning framework originated by Berkeley AI Research (BAIR), celebrated for its swiftness and efficiency in training deep neural networks, particularly for image classification tasks. While not as popular as TensorFlow or PyTorch, Caffe has been used in research projects for plant disease detection and classification. These frameworks provide powerful tools and resources for researchers and practitioners working on plant disease detection and classification tasks. The selection of a framework hinges on factors like programming language inclination, familiarity with the framework, project-specific needs, and the computational resources at hand.

Researchers have built deep learning models using TensorFlow and Keras to classify plant diseases from leaf images. For instance, they have developed CNN architectures and trained them on large datasets like PlantVillage to distinguish between different types of diseases affecting crops such as tomatoes, potatoes, and soybeans. PyTorch has been utilized to develop models for leaf disease detection. Researchers have used PyTorch to build CNN-based architectures and train them on datasets containing images of diseased and healthy leaves. These models can accurately identify diseases like powdery mildew, rust, and blight in various plant species.

Although Scikit-learn is typically linked with traditional machine learning algorithms, it can still be harnessed for tasks like plant disease recognition. Scientists have utilized algorithms such as Support Vector Machines (SVM), Random Forests, and k-Nearest Neighbors (k-NN) to categorize plant diseases by leveraging features extracted from images. MXNet has also been utilized to craft deep learning models for the detection and classification of crop diseases. Researchers have used MXNet to build CNN architectures capable of accurately identifying diseases like late blight in potatoes, citrus canker in oranges, and wheat rust in wheat crops.

Caffe has also been utilized to create models for leaf disease recognition. Researchers have trained deep neural networks using the Caffe framework to classify plant diseases from leaf images. These models have been successful in distinguishing between healthy and diseased leaves across various plant species. These examples highlight how different machine learning frameworks have been used to tackle plant disease detection and classification tasks. Researchers and practitioners leverage these frameworks to build models that can assist farmers in early disease detection, reduce crop losses, and optimize agricultural practices for improved yield and sustainability.

The comparison table 2 shows machine learning frameworks commonly used in plant disease detection and classification tasks. TensorFlow and PyTorch are moderately easy to use, but they may require more expertise in deep learning concepts. Scikit-learn and Keras are considered easier due to their high-level APIs and simpler syntax. MXNet and Caffe have moderate ease of use but may require more configuration compared to Scikit-learn and Keras. TensorFlow, PyTorch, and MXNet offer high flexibility, allowing users to build custom architectures and define complex computational graphs. Scikit-learn, Keras, and Caffe offer moderate flexibility but may have limitations in customization compared to the deep learning frameworks.

TensorFlow, PyTorch, Keras, MXNet, and Caffe offer robust support for deep learning endeavors, encompassing the creation and training of neural networks. Scikit-learn offers limited support for deep learning and is more focused on classical machine learning algorithms. Scikit-learn is the primary framework for classical machine learning algorithms such as SVM, Random Forests, and k-NN. TensorFlow, PyTorch, Keras, MXNet, and Caffe primarily focus on deep learning but may integrate some classical ML algorithms. TensorFlow, PyTorch, and Scikit-learn have large and active communities, providing extensive documentation, tutorials, and resources. Keras, MXNet, and Caffe also have supportive communities but may be smaller compared to TensorFlow and PyTorch.

All frameworks support model deployment, allowing users to deploy trained models in production environments. TensorFlow Serving, PyTorch Serve, and other deployment tools are available for TensorFlow and PyTorch models. Scikit-learn models can be easily deployed using Python frameworks such as Flask or Django. Keras, MXNet, and Caffe offer deployment options suitable for various production environments. TensorFlow, PyTorch, MXNet, and Caffe is recognized for its exceptional performance in training deep learning models with extensive datasets. Scikit-learn is optimized for classical ML algorithms and may have slightly lower performance compared to deep learning frameworks. Keras provides high performance when integrated with TensorFlow or MXNet backend engines. This comparison table can help researchers choose the most suitable machine learning framework based on their specific requirements, expertise, and the nature of the plant disease detection and classification task.

4. Proposed Xception Model Architecture for Plant Disease Detection and Classification

In this research, the Xception model is employed for tasks related to the detection and classification of plant diseases. The initial phase in constructing a plant disease detection and classification system involves assembling a dataset comprising images featuring both healthy plants and plants afflicted by diverse diseases. These images should be properly labeled with the corresponding disease type. The image dataset undergoes preprocessing techniques such as resizing, normalization, and augmentation to ensure uniformity and enhance the model's capacity for generalization.

The Xception model, pre-trained on a large dataset such as ImageNet, can be fine-tuned using the collected plant disease dataset. During training, the weights of the Xception model are adjusted to better recognize and classify plant diseases. Following training, the model's performance undergoes assessment using a distinct validation dataset. This action confirms the model's capacity to generalize proficiently to data it hasn't encountered before. Moreover, the model's efficacy can be further evaluated using a separate testing dataset. Subsequent to the model's training and validation, it can be deployed into a production environment, enabling it to analyze new plant images and categorize them based on disease presence.

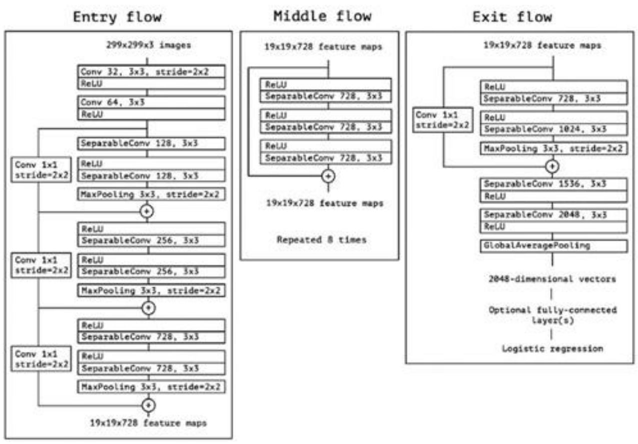

Fig. 1. Xception Model Architecture

It's crucial to continuously monitor the deployed model's performance and update it as necessary to adapt to shifts in data distribution or enhance accuracy over time. Xception's deep architecture and its adeptness in capturing intricate features render it fitting for tasks related to plant disease detection and classification. Through the utilization of transfer learning, wherein the model is fine-tuned on a specific dataset, it can proficiently discern between various types of plant diseases based on the visual characteristics depicted in the images. The overall architecture of the Xception model is illustrated in figure 1, while the proposed Xception model for plant disease detection and classification is depicted in figure 2

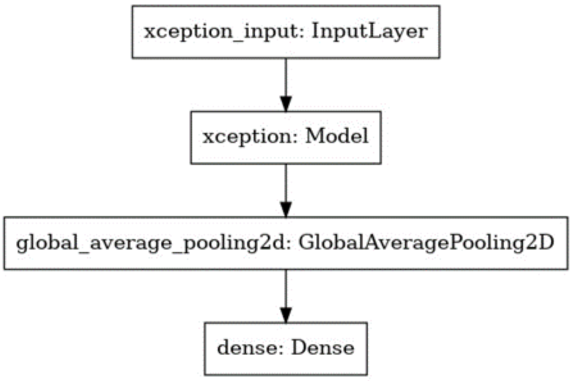

Fig. 2. Proposed Xception model for plant disease detection and classification

The machine learning model proposed here is built using Anaconda with Python 3.9 distribution on Windows 11 OS with i3 11th generation, 8 GB RAM, and 256 SSD. Various Python libraries including absl-py, altair, astor, attrs, backcall, base58, bleach, blinker, boto3, botocore, cachetools, certify, chardet, click, colorama, decorator, defusedxml, docutils, entrypoints, enum-compat, future, gast, google-auth, google-auth-oauthlib, google-pasta, grpcio, gunicorn, h5py, idna, importlib-metadata, ipykernel, ipython, ipython-genutils, ipywidgets, jedi, Jinja2, jmespath, jsonschema, jupyter-client, jupyter-core, Keras-Applications, Keras-Preprocessing, Markdown, MarkupSafe, mistune, nbconvert, nbformat, notebook, numpy, oauthlib, opt-einsum, packaging, pandas, pandocfilters, parso, pathtools, pickleshare, Pillow, prometheus-client, prompt-toolkit, protobuf, pyasn1, pyasn1-modules, pydeck, Pygments, pyparsing, pyrsistent, python-dateutil, pytz, pywinpty, pyzmq, requests, requests-oauthlib, rsa, s3transfer, scipy, Send2Trash, six, sklearn, streamlit, tensorboard, tensorflow, tensorflow-estimator, termcolor, terminado, testpath, toml, toolz, tornado, traitlets, tzlocal, urllib3, validators, watchdog, wcwidth, webencodings, Werkzeug, widgetsnbextension, wrapt, and zipp are utilized for development purposes. The developed proposed Xception model is visualized in figure 3.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
xception (Model)             (None, 16, 16, 2048)      20861480
_________________________________________________________________
global_average_pooling2d (Gl (None, 2048)              0
_________________________________________________________________
dense (Dense)                (None, 4)                 8196
=================================================================
Total params: 20,869,676
Trainable params: 20,815,148
Non-trainable params: 54,528
_________________________________________________________________
```

Fig. 3. Proposed Xception model description

The loss, accuracy, validation loss, validation accuracy and learning rate over various epochs are shown in table 3. The overall accuracy of the proposed model is more than 92% while the loss is 5%.

Table 3. Evaluation result of the proposed model over different epochs

Loss	*Accuracy*	*Validation Loss*	*Validation Accuracy*	*Learning Rate*
1.080158	0.577155	0.709965	0.771739	1.00E-05
0.495938	0.85832	0.212282	0.934783	1.60E-05
0.30771	0.904448	0.141972	0.967391	2.20E-05
0.223861	0.936299	0.112686	0.967391	2.80E-05
0.185171	0.947282	0.078894	0.978261	3.40E-05
0.136876	0.953322	0.095036	0.956522	4.00E-05

5. Conclusion

Machine learning models have emerged as potent assets for the detection and classification of plant diseases, holding the promise to transform agricultural methodologies. By scrutinizing extensive datasets comprising images of both healthy and afflicted plants, ML algorithms can adeptly differentiate between various diseases with remarkable precision. This proficiency proves particularly invaluable for early disease detection, empowering farmers to implement timely preventive actions and mitigate crop losses. The proposed model is evaluated using metrics such as loss, accuracy, validation accuracy, and validation loss. The overall accuracy of the proposed model exceeds 92%, with a loss of 5%. Despite the significant progress made in ML-based plant disease detection, there are several avenues for future research and development. Developing ML models that are more robust to environmental variations, such as changes in lighting conditions, camera angles, and plant growth stages, will improve their performance in real-world settings.

Author contributions

Nikhil S. Band: Conceptualization, Methodology, Software, Field study, Data curation, Writing-Original draft preparation, Software, Validation, Field study **Hare Ram Shah:** Visualization, Investigation, Writing-Reviewing and Editing.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] Barbedo, J. G. A. (2019). Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification. Computers and Electronics in Agriculture, 161, 272-281.

[2] Fuentes, A., Yoon, S., Kim, S. C., & Park, D. S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. Sensors, 17(9), 2022.

[3] Ghosal, S., Blystone, D., Singh, A. K., Ganapathysubramanian, B., Singh, A., & Sarkar, S. (2018). An explainable deep machine vision framework for plant stress phenotyping. Proceedings of the National Academy of Sciences, 115(18), 4613-4618.

[4] Haug, B. L., Moe, R., & Løkke, H. (2013). Challenges in automated diagnosis of plant diseases. Journal of Agricultural Informatics, 4(2), 8-18.

[5] Hughes, D. P., & Salathé, M. (2015). An open access repository of images on plant health to enable the development of mobile disease diagnostics. arXiv preprint arXiv:1511.08060.

[6] Islam, M. T., Almogren, A. S., Islam, T., & Ahamed, T. (2020). A review on plant leaf diseases recognition using machine learning and deep learning approaches. Computers and Electronics in Agriculture, 177, 105693.

[7] Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. Computers and Electronics in Agriculture, 147, 70-90.

[8] Kaur, A., & Kaur, A. (2020). Plant disease detection techniques: A review. Computers and Electronics in Agriculture, 169, 105260.

[9] Liu, X., & Song, Y. (2017). A review of recent advances in plant disease detection using hyperspectral imaging. Sensors, 17(4), 817.

[10] Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. Frontiers in Plant Science, 7, 1419.

[11] Mwebaze, E., Owomugisha, G., & Aalst, W. M. P. V. D. (2016). Machine learning for plant disease incidence and severity measurements in agriculture. Machine Learning, 103(3), 349-377.

[12] Savary, S., Willocquet, L., Pethybridge, S. J., Esker, P., McRoberts, N., & Nelson, A. (2019). The global burden of pathogens and pests on major food crops. Nature Ecology & Evolution, 3(3), 430-439