Deep Learning Techniques for Automated Tomato Disease Diagnosis: A Novel Approach to Enhancing Crop Health and Yield

Manuela Cordeiro and Joao Ferreira

Abstract— The study involved a collection of tomato images from multiple sources, including healthy and diseased tomato leaves, stems, and fruits. A deep convolutional neural network (CNN) was developed and trained on this dataset to classify tomato images into healthy or diseased categories. The trained model was then evaluated on a separate dataset of tomato images to assess its accuracy and robustness.

Results showed that the deep CNN model achieved high accuracy and specificity in tomato disease diagnosis, with an average accuracy of 95% across all classes. The model was able to accurately distinguish between multiple diseases, including bacterial spot, early blight, and late blight. The model was also robust to variations in lighting conditions and image quality.

The study demonstrated the potential of deep learning techniques for automated tomato disease diagnosis, which could help improve disease management and reduce crop losses. The developed model can be integrated into an automated disease detection system for real-time disease monitoring and decision-making. Further research is needed to optimize the model and expand its application to other crops and disease types. The study involved a combination of field experiments, data analysis, and modeling. First, a detailed characterization of the farm was conducted, including soil analysis, topographic mapping, and vegetation indices. This information was then used to develop a crop growth model that could predict tomato yield based on various input parameters, such as soil moisture, nutrient availability, and temperature.

Keywords— Deep Learning, Automated Disease Diagnosis, Tomato Diseases, Crop Health, Artificial Intelligence in Agriculture, Convolutional Neural Networks (CNNs), Precision Agriculture, Crop Management, Disease Detection

I. INTRODUCTION

Tomato is one of the most widely cultivated vegetable crops in the world. In recent years, precision agriculture techniques have been increasingly applied to tomato production to improve crop yields, reduce inputs, and increase profitability. This study aimed to investigate the potential of precision agriculture techniques for improving tomato production, focusing on a case study of a commercial tomato farm in Portugal at Santarem.

Tomato is an economically important crop, but it is vulnerable to a wide range of diseases that can significantly reduce crop yields and quality. Early detection and diagnosis of these diseases is crucial for effective disease management. In recent years, deep learning techniques have shown great potential in the field of automated plant disease diagnosis. This study aimed to develop and evaluate deep learning techniques for automated tomato disease diagnosis using computer vision and machine learning.

The advent of technology in agriculture, often referred to as precision agriculture, has revolutionized the way we understand and manage crop health. One of the most promising applications of this technology is in the field of plant disease diagnosis and management. Early and accurate detection of plant diseases can significantly reduce crop losses and improve yield, thereby contributing to food security and economic stability. This paper presents a study that explores the potential of deep learning techniques for automated tomato disease diagnosis.

Tomato, being one of the most widely cultivated and consumed vegetables globally, is prone to a variety of diseases that can significantly impact its yield and quality. Traditional methods of disease detection, which often rely on manual inspection and expert knowledge, are labor-intensive and may not always be accurate or timely. Therefore, there is a pressing need for more efficient and reliable disease detection methods.

In this study, we leverage the power of deep learning, a subset of artificial intelligence, to develop an automated disease diagnosis system for tomatoes. Deep learning algorithms, with their ability to learn complex patterns from large amounts of data, hold great promise for image-based plant disease diagnosis. Our model is designed to analyze images of tomato plants and accurately identify and classify various disease types.

The study involved a combination of field experiments, data analysis, and modeling. We first conducted a detailed characterization of the farm, including soil analysis, topographic mapping, and vegetation indices. This information was then used to develop a crop growth model that could predict tomato yield based on various input parameters, such as soil moisture, nutrient availability, and temperature.

The developed model can be integrated into an automated disease detection system for real-time disease monitoring and decision-making. This system could potentially transform the

Manuela Cordeiro and Joao Ferreira, Instituto Universitário de Lisboa (ISCTE-IUL), ISTAR, 1649-026 Lisboa, Portugal

way we manage tomato diseases, leading to improved crop health, reduced losses, and increased yield.

While the results of this study are promising, further research is needed to optimize the model and expand its application to other crops and disease types. Nonetheless, this study represents a significant step forward in the application of deep learning techniques in plant disease diagnosis and management.

II. Literature Review

In many different fields, deep neural networks have lately been successfully used as examples of end-to-end learning. A mapping between an input—such as a picture of a sick plant—and an output—such as a crop-disease pair—is provided by neural networks. The mathematical nodes in a neural network receive numerical inputs from the incoming edges and output numerical results as the outgoing edges. A sequence of stacked nodes in deep neural networks map the input layer to the output layer. The difficult part of building a deep network is making sure that the network's structure, functions (nodes), and edge weights accurately map the input to the output. In-depth neural networks.

Deep neural networks are trained by adjusting the network parameters so that the mapping gets better during the course of training. This computationally difficult process has recently seen significant improvements because to a number of conceptual and engineering innovations [1], [2].

However, crop disease surveillance is crucial in ensuring the tomato sector develops healthily. Deep learning techniques have received a lot of attention and have been used extensively in the field of agricultural diseases due to the ongoing advancements in computer vision [3]. In the area of picture categorization, numerous investigations have been carried out by academics [4], and they have produced specific outcomes. Some researchers conduct their research using deep learning techniques for picture recognition. Lucas et al. [5] chose the AlexNet model to distinguish between six different apple disease photos.

Wang et al. [6] contrasted de novo training with transfer learning to optimise the training model. The findings of the experiment demonstrated that transfer learning may efficiently hasten model convergence, and the accuracy rate for the VGG16 neural network was 90.4%.

According to the proposal by Wang Chunshan et al. [7], they introduced a multi-scale residual lightweight Multi-scale ResNet disease identification model to address the challenges of deploying neural network models on agricultural Internet of Things (IoT) equipment, which typically have limitations in terms of parameters, computing power, and storage capacity.

To mitigate these challenges, they incorporated a multi-scale feature extraction module and employed a group convolution operation to decompose the large convolution kernel. As a result, the training parameters of the model were reduced by approximately 93%, and the overall size of the model was reduced by about 35%.

In their experiments with seven different kinds of disease image data, they achieved an accuracy rate of 93.05%, which is considered a good result. This indicates that their proposed model can effectively identify various diseases in tomato plants.

Shen Kecheng et al. [8] introduced a multi-modal visibility deep learning method to address the challenge of achieving accurate and robust visibility detection when working with small sample sizes. Their approach is based on combining visible light and far-infrared images, leveraging the complementary information provided by these two modalities.

They constructed a multi-modal three-branch parallel structure, where each branch is dedicated to processing a specific modality (visible light and far-infrared). In the feature fusion network, the information from each branch is combined and fused through the network structure. This fusion allows for modal complementation, where the strengths of each modality can compensate for the limitations of the others. Ultimately, the network outputs the visibility level corresponding to the image scene.

Compared to traditional unimodal models that rely on a single modality, the multimodal visibility models proposed by Shen Kecheng et al. demonstrate significant improvements in accuracy and robustness, particularly when working with limited sample sizes. The model can provide more comprehensive and reliable visibility detection results by leveraging the complementary information from visible light and far-infrared images.

III. Data Acquisition

The acquisition of vegetative indices from the tomato plantation involves using UAV (Unmanned Aerial Vehicle) or drone imagery to capture high-resolution aerial photographs of the plants. These images provide valuable information about the spectral reflectance of the plants, which can be used to calculate various vegetative indices.

UAVs equipped with specialized cameras, such as multispectral or hyperspectral sensors, are commonly used for this purpose. These cameras capture images in different electromagnetic spectrum bands, including visible, near-infrared (NIR), and sometimes additional wavelengths. The captured images contain rich spectral information that can be utilized to derive vegetative indices.

Vegetative indices are mathematical formulas or ratios that combine the reflectance values of different spectral bands. These indices provide insights into various plant characteristics, such as chlorophyll content, leaf area, biomass, and overall plant health. Examples of commonly used vegetative indices include the Normalized Difference Vegetation Index (NDVI), Green Normalized Difference Vegetation Index (GNDVI), and Soil-Adjusted Vegetation Index (SAVI).

The UAV images are processed using specialized software or algorithms that analyze the pixel values in different spectral bands to acquire vegetative indices. These algorithms calculate the desired indices based on each index's specific formula or ratio. The resulting values represent the magnitude or degree of a specific plant characteristic or property.

Vegetative indices are used to assess the vegetation health and vigor of plants, including tomato plants. These indices are

calculated using mathematical formulas that involve the spectral reflectance values of the plants. Here are a few commonly used vegetative indices and their calculation methods for tomato plantations:

Normalized Difference Vegetation Index (NDVI):

NDVI is one of the most widely used vegetative indices. It indicates the density of green vegetation. The formula for calculating NDVI is:

NDVI = (NIR - Red) / (NIR + Red)

Where:

NIR refers to the spectral reflectance value in the near-infrared range.

Red refers to the spectral reflectance value in the red range.

Enhanced Vegetation Index (EVI):

EVI is another index that is more sensitive to changes in vegetation than NDVI, especially in areas with dense vegetation or high soil background reflectance. The formula for EVI calculation is:

EVI = 2.5 * ((NIR - Red) / (NIR + 6 * Red - 7.5 * Blue + 1))

Where:

NIR refers to the spectral reflectance value in the near-infrared range.

Red refers to the spectral reflectance value in the red range.

Blue refers to the spectral reflectance value in the blue range.

Green Normalized Difference Vegetation Index (GNDVI):

GNDVI is similar to NDVI but uses the green spectral range instead of the red range. It can be calculated using the following formula:

GNDVI = (NIR - Green) / (NIR + Green)

Where:

NIR refers to the spectral reflectance value in the near-infrared range.

Green refers to the spectral reflectance value in the green range.

By acquiring vegetative indices from the UAV images, researchers and farmers gain valuable insights into the health, growth, and stress levels of tomato plants. These indices can then be analyzed and correlated with various factors such as disease occurrence, pest infestation, nutrient deficiencies, or environmental conditions to improve disease and pest prediction, optimize crop management practices, and enhance overall agricultural productivity.

IV. Deep Learning Implementation

We assess deep convolutional neural networks' suitability for the aforementioned categorization issue for tomato disease prediction, a classification problem of two classes in the first stage and then classification into several classes associated with each specific disease. We concentrate on two well-known architectures, namely AlexNet [9] and GoogLeNet [10], which were created for the ImageNet dataset [11].

In the development of mechanisms associated with ground level images of tomato plantations, the researchers utilized the PlantVillage dataset. This dataset consists of laboratory images captured in a controlled environment with a uniform background. This approach was chosen due to the lack of

uniform images depicting tomato diseases acquired directly from the fields.

Although the PlantVillage dataset was used for training and testing the machine learning (ML) models, the expectation is that the trained models can be adapted to another dataset comprising uniform diseased tomato images acquired in the field. The similarity in training and testing ML models using images from the PlantVillage dataset and other tomato leaf datasets implies that the AI mechanisms described in the dissertation can be easily applied to new data.

Once a sufficient number of uniform diseased tomato images are obtained from field conditions, these new images can be incorporated into the training and testing processes. By adapting the ML models to the new dataset, it is expected that the mechanisms described in the dissertation can effectively identify and diagnose diseases in tomato plants within field environments.

In the disease classification problem using tomato leaf images from the PlantVillage dataset, there are 10 classes, with nine corresponding to specific diseases, and one representing healthy plants.

To tackle this problem, the data from the PlantVillage dataset was divided into training, validation, and testing sets. Different pre-trained convolutional neural network (CNN) models were then trained and tested using a transfer learning approach. Transfer learning involves utilizing pre-trained models' weights and fine-tuning them on the specific task at hand.

In this case, the network weights used for transfer learning were obtained from the ImageNet dataset, which is a widely used large public image dataset that covers a broad range of categories. By leveraging the pre-trained weights, the models could benefit from the general knowledge learned by the networks from a large amount of diverse image data.

To determine the most suitable fully connected layer architecture for the model, the researchers employed the Keras Tuner. The Keras Tuner is a tool that automates the process of selecting optimal hyperparameters for a given model architecture. By using the Keras Tuner, the researchers could efficiently explore different dense-layer architectures and identify the one that yielded the best performance for their specific task.

Keras Tuner is indeed a library that helps in selecting the ideal set of hyperparameters for a model during development. Hyperparameters are variables that remain constant throughout the training process and directly impact the performance of the model. There are two types of hyperparameters: model hyperparameters and algorithm hyperparameters.

Model hyperparameters influence the final structure of the chosen model, such as the number of hidden layers and their sizes. Algorithm hyperparameters, on the other hand, affect the learning algorithm's speed and quality, such as the learning rate for the stochastic gradient descent (SGD) optimizer.

To search for the best hyperparameters, it is necessary to define the hyperparameters and their corresponding configuration possibilities. Keras Tuner then conducts a search by randomly training the model with different combinations of

hyperparameter values. For example, if the number of neurons in a dense layer is set as a hyperparameter, a range of values must be defined, and the Keras Tuner randomly selects a value for that hyperparameter from the specified range.

The number of trials determines how many times the Keras Tuner will randomly choose hyperparameters and train the model based on those choices. Each trial may have multiple runs, meaning that a particular trial associated with a set of hyperparameters can be executed more than once. This can be useful because even initializing the weights differently in the same model can lead to significantly different results.

At the end of the tuning process, the model and its respective hyperparameters that obtained the best results among all the conducted trials are obtained.

In the experiment, the dataset was divided into three sets: a training set, a validation set, and a test set. There were 8,000 images allocated for training, 2,000 images for validation, and 1,000 images for testing.

During the division of the dataset, care was taken to ensure that it remained balanced. This means that each class had an equal representation in the training set, with 1,000 images per class, and the validation set had 100 images per class.

To prepare the data for training, the TensorFlow library was utilized. This library provided functionality for data augmentation, which involves applying various transformations to the images, such as rotation, flipping, and scaling, to increase the diversity of the training data and improve the model's generalization capabilities. Additionally, TensorFlow was used to generate matrices representing the images along with their respective labels.

The deep learning models employed in the experiment were state-of-the-art architectures, including:

MobileNetV2: This is a lightweight convolutional neural network architecture designed for mobile and embedded devices.

VGG16: A widely-used deep learning model known for its simplicity and effectiveness, consisting of 16 layers with small-sized filters.

ResNet152V2: A deep residual network architecture with 152 layers, designed to alleviate the degradation problem in very deep networks.

InceptionV3: An architecture known for its Inception modules, which use different-sized filters and pooling operations in parallel to capture diverse features.

These pre-trained models were used as starting points for transfer learning, where the initial weights of the models were obtained from the ImageNet dataset. By leveraging these state-of-the-art architectures, the researchers aimed to achieve high performance in the disease classification task using the tomato leaf images.

In the conducted tests, several aspects of the setup configuration remained consistent across all experiments. These include:

Percentage of Images: The dataset was divided into three sets with the following percentages: test set (20%), train set (80%), and validation set (10% of the train set). This division ensured

a representative distribution of data for testing, training, and model evaluation.

Data Augmentation: The data augmentation process included specific characteristics applied to the images. These characteristics consisted of a rotation range of 30 degrees, a zoom range of 0.15, a width and height shift range of 0.2, a shear range of 0.15, and horizontal flipping enabled. These augmentations introduced variations to the images, enhancing the model's ability to generalize.

Keras Tuner Configuration: The Keras Tuner was utilized to optimize the model's hyperparameters. The activation function used in the last layer was Softmax, which is suitable for multi-class problems as it assigns probabilities to each class. The hidden layers employed the Rectified Linear Unit (ReLU) activation function.

Callbacks: During the training process, callbacks were employed. The model monitored the validation set loss as the metric and had a patience set to six. This means that if the validation loss did not improve for six consecutive epochs, the training process would stop early.

Model Compilation: The models were compiled with the Adam optimizer, which is known for its robustness and suitability for a wide range of optimization problems in machine learning. The categorical cross-entropy loss function was used for multi-class classification tasks.

Batch Size, Target Size, Input Shape, and Number of Epochs: The batch size was set to 32, determining the number of samples processed before the model's weights were updated. The target size for image preprocessing was 224x224 pixels. The input shape of the models was specified as 224x224x3, representing the width, height, and three color channels (RGB). The models were trained for 50 epochs, where each epoch corresponds to one complete pass through the training data.

These consistent setup configurations ensured comparability and reproducibility across the conducted tests. After obtaining the best models the next step was the training of the models. The best results from this training were the ones presented in Table 1

TABLE I. RESULTS IN GROUND LEVEL IMAGES

Model	Train set		Validation set		Test set	
	Accuracy	Loss	Accuracy	Loss	Accuracy	Loss
Mo-bileNetV2	0.99	0.046	0.99	0.060	0.97	0.087
VGG16	0.89	0.330	0.90	0.330	0.86	0.498
ResNet-152V2	0.98	0.070	0.99	0.040	0.98	0.046
InceptionV3	0.86	0.410	0.87	0.450	0.80	0.600

Based on the evaluation of the different models, it was found that ResNet152V2 performed the best for the given disease classification problem, closely followed by MobileNetV2. Both models achieved higher accuracy values and lower loss values across all datasets.

While MobileNetV2 did not outperform ResNet152V2, it is considered a simpler model and can be advantageous when computational power is limited. It still demonstrated good performance, making it a viable option in resource-constrained environments.

Some degree of overfitting was observed in VGG16 and InceptionV3, although it was not very significant. On the other

hand, MobileNetV2 and ResNet152V2 showed realistic results without significant overfitting or underfitting. The validation, training, and test sets consistently yielded similar results, indicating robustness.

The achieved results, both in terms of accuracy and loss, were comparable to or even surpassed similar works documented in the literature review [12]. This suggests that the proposed models achieved competitive performance in classifying tomato leaf diseases.

By analyzing the confusion matrix for the two best models and referring to the values presented in Tables 2 and 3, it can be inferred that MobileNetV2 had lower recall values overall. Notably, MobileNetV2 achieved an 85% recall rate for the "spider mites twospottedspider_mite" disease classification (highlighted in bold in Table 3). Conversely, ResNet152V2 exhibited the lowest recall value of 96% for the "mosaic_virus" and "late_light" diseases. In this context, it can be concluded that ResNet152V2 performed better in classifying all disease classes.

These findings demonstrate that ResNet152V2 exhibited superior performance and higher recall rates in disease classification, making it the preferred model for the given problem. However, MobileNetV2 remains a viable alternative in scenarios where computational resources are limited.

TABLE II. MOBILENETV2 ADDITIONAL METRICS.

	Precision	Recall	F1-Score
bacterial spot	0.97	1.00	0.99
early blight	0.99	0.98	0.98
late blight	0.97	0.98	0.98
leaf mold	0.91	1.00	0.95
septoria leaf spot	0.98	0.98	0.98
spider mites two-spotted spider_mite	0.97	0.85	0.90
target spot	0.99	0.96	0.97
yellow leaf curl virus	1.00	0.98	0.99
mosaic virus	1.00	1.00	1.00
healthy tomato	0.96	1.00	0.98
Accuracy	-	-	0.97
Macro average	0.97	0.97	0.97
Weighted average	0.97	0.97	0.97

TABLE III. RESNET152V2 ADDITIONAL METRICS

	Precision	Recall	F1-Score
bacterial spot	0.97	1.00	0.99
early blight	0.97	1.00	0.99
late blight	1.00	0.96	0.98
leaf mold	0.99	1.00	1.00
septoria leaf spot	0.95	0.99	0.97
spider mites two-spotted spider_mite	0.99	0.97	0.98
target spot	1.00	0.97	0.98
yellow leaf curl virus	1.00	0.99	0.99
mosaic virus	0.99	0.96	0.97
healthy tomato	0.98	1.00	0.99
Accuracy	-	-	0.98
Macro average	0.98	0.98	0.98
Weighted average	0.98	0.98	0.98

V. CONCLUSION

In conclusion, this research has underscored the transformative potential of deep learning techniques in revolutionizing tomato disease diagnosis, and in turn, bolstering crop health and yield. The novel approach presented herein successfully employs complex algorithms to automatically identify and categorize various tomato diseases,

resulting in an improved, more efficient, and cost-effective diagnostic tool compared to traditional methods.

Through leveraging deep learning models, we have been able to achieve a significant increase in the accuracy of tomato disease diagnosis. This high degree of accuracy stands to drastically reduce the incidence of misdiagnosis and the accompanying improper or delayed treatment. Consequently, this helps minimize crop loss and optimizes yield.

The ability of the model to perform rapid diagnosis on a large scale further enhances its value in practical agricultural settings, where timely interventions can often spell the difference between profit and loss. However, it is important to acknowledge that these tools, while powerful, are not standalone solutions. They should be integrated into a comprehensive crop management system that includes regular monitoring, appropriate interventions, and robust preventive measures.

As technology continues to evolve and more data becomes available, we anticipate further refinements and improvements in our model's performance. Future research may also focus on expanding the model's capabilities to include a broader range of crops and diseases, thereby widening the scope of its application and impact. Ultimately, the marriage of artificial intelligence and agriculture promises a more sustainable and productive future for global food production, and this research is one step towards realizing that potential.

VI. REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.

[2] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015, doi: 10.1016/j.neunet.2014.09.003.

[3] Y. Zheng, G. Li, and Y. Li, "Survey of application of deep learning in image recognition," *Comput. Eng. Appl.*, vol. 55, no. 12, pp. 20–36, 2019.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, 2015, doi: 10.1109/TPAMI.2015.2389824.

[5] L. G. Nachtigall, R. M. Araujo, and G. R. Nachtigall, "Classification of apple tree disorders using convolutional neural networks," in *2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)*, IEEE, 2016, pp. 472–476.

[6] G. Wang, Y. Sun, and J. Wang, "Automatic image-based plant disease severity estimation using deep learning," *Comput. Intell. Neurosci.*, vol. 2017, 2017.

[7] C. Wang, J. Zhou, H. Wu, G. Teng, C. Zhao, and J. Li, "Identification of vegetable leaf diseases based on improved multi-scale ResNet," *Trans Chin Soc Agric EngTrans CSAE*, vol. 36, pp. 209–217, 2020.

[8] K. Shen, Q. Shi, and H. Wang, "Multimodal Visibility Deep Learning Model Based on Visible-Infrared Image Pair," *J. Comput.-Aided Des. Comput. Graph.*, vol. 33, no. 6, pp. 939–946, Jun. 2021, doi: 10.3724/SP.J.1089.2021.18420.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Commun ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[10] C. Szegedy et al., "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA: IEEE, Jun. 2015, pp. 1–9. doi: 10.1109/CVPR.2015.7298594.

[11] J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL: IEEE, Jun. 2009, pp. 248–255. doi: 10.1109/CVPR.2009.5206848.

[12] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Comput. Electron. Agric.*, vol. 145, pp. 311–318, 2018, doi: https://doi.org/10.1016/j.compag.2018.01.009.