International Research Journal of Education and Technology
Peer Reviewed Journal
ISSN 2581-7795

Image Based Crop Classification using DNN

Rithani Bakthavatchalam
Department of Biotechnology
Bannari Amman Institute of Technology
Sathyamangalam, Erode
rithani.bt19@bitsathy.ac.in

Nagavarsini K
Department of Agriculture
Bannari Amman Institute of Technology
Sathyamangalam, Erode
nagavarsini.ag19@bitsathy.ac.in

Harini S
Department of Agriculture
Bannari Amman Institute of Technology
Sathyamangalam, Erode
harini.ag19@bitsathy.ac.in

Balasamy K
Department of Artificial Intelligence and Data science
Bannari Amman Institute of Technology
Sathyamangalam, Erode
balasamyk@bitsathy.ac.in

***Abstract*— The latest generation of convolutional neural networks (CNNs) has achieved impressive results in the field of image classification. This paper is concerned with a new approach to the development of plant disease recognition model, based on leaf image classification, by the use of deep convolutional networks. Novel way of training and the methodology used facilitate a quick and easy system implementation in practice. The developed model is able to recognize different types of plant diseases out of healthy leaves, with the ability to distinguish plant leaves from their surroundings. According to our knowledge, this method for plant disease recognition has been proposed for the first time. All essential steps required for implementing this disease recognition model are fully described throughout the project, starting from gathering images in order to create a database, assessed by agricultural experts. Neural network, was used to perform the disease prediction. The experimental results on the developed model achieved precision for separate class tests, on average.

Keywords—Neural network , Database, Disease detection

I. Introduction

Despite significant advancements in the mass production and accessibility of food, a number of variables, like the dwindling pollinator population and plant diseases, continue to pose a danger to food security. More than 80% of agricultural production in the developing world is produced by smallholder farmers, and estimates of yield losses of more than 50% due to pests and diseases are frequent. In addition, the vast majority of people who experience hunger reside in smallholder agricultural households. Fortunately, infections may be controlled by spotting them as soon as they show up on a plant. Additionally, it's simple to acquire information on a certain type of disease's diagnosis because to the global spread of the internet and mobile technology. Therefore, the proliferation of smartphones with powerful cameras can aid in scaling up any type of crop identification method to make it feasible and useful. Because of their computer power, high-resolution displays, and comprehensive built-in accessory sets, such as powerful HD cameras, smartphones in particular offer very unique techniques to aid in the identification of diseases. In fact, it is predicted that by the year 2050, there will be almost 6 billion phones on the market. The algorithm in this paper will be fed 2D photos of healthy and damaged plant leaves. I'll be utilising a deep convolutional network, a generative adversarial network, and a ladder network-based semi-supervised learning strategy. These many methods will be employed to produce a predicted illness type or a type of condition.

A. *Limitation of existing system:*

The current method for identifying plant diseases relies solely on naked eye observation, which demands additional manpower, adequately outfitted labs, pricey gadgets, etc. And incorrect disease identification may have resulted in inexperienced pesticide application, which may have led to the development of long-term pathogen resistance, decreasing the ability of the crop to defend itself. The precision of the system is low and processing time for existing approaches is lengthy.

B. *Proposed System:*

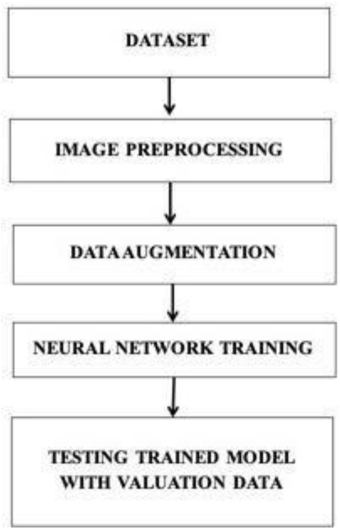

Fig-1 Methodology flowchart

A plant disease detection system based on convolutional neural networks (CNNs) that makes more accurate disease predictions has been proposed.

II. PROCESS IN CNN:

A. Dataset collection:

At every stage of object recognition research, from the training phase to assessing the effectiveness of recognition algorithms, appropriate datasets are necessary. The Internet was used to download all of the photographs used in the collection, and different sources in different.

B. Image Preprocessing and Labelling:

The resolution and quality of the images I acquired from the Internet varied along with the formats they were in. Final photos that were going to be utilized as a dataset for a deep neural network classifier were preprocessed to improve feature extraction. Additionally, each image was manually cropped as part of the image preparation process in order to highlight the region of interest.

C. Augmentation Process:

The basic goal of using augmentation is to expand the dataset and somewhat distort the images, which aids in lowering overfitting during the training stage. Image data augmentation is a method for artificially increasing the size of a training dataset by producing altered versions of the dataset's photographs.

More data can be used to train deep learning neural network models, which can lead to more proficient models. Additionally, by using augmentation techniques to change the original photos, fit models are better able to apply what they have learnt to new images.

D. Neural Network Training:

The primary objective of network training is to teach the neural network the characteristics that set one class apart from the others. Therefore, the likelihood that the network will learn the proper features has grown by using more enhanced photos.

E. Testing Trained Model With Valuation Data:

The trained network is then utilised to process the input photographs in the valuation dataset in order to detect the disease, and the results are processed.

III. SOFTWARE AND SYSTEM SPECIFICATION THE TEMPLATE

A. Anaconda (Python distribution)

Anaconda is a free and open-source distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), With the goal of streamlining package management and deployment. Conda, a package management system, controls package versioning. For Windows, Linux, and MacOS, the Anaconda distribution provides data-science packages.

The Anaconda distribution includes the Conda package management and Virtual Environment Manager in addition to more than 1,500 packages. A GUI, named Anaconda Navigator, is also included as a graphical replacement for the command line interface (CLI). The management of package dependencies is a significant difficulty for Python data science and the main distinction between conda and the pip package manager. This is the basis for conda's existence. Without determining if they conflict with already-installed packages, dependent Python packages are immediately installed by pip when a package is installed.

B. Anaconda Navigator

Without using command-line commands, users can start programmes and manage conda packages, environments, and channels with the help of Anaconda Navigator, a desktop GUI that is part of the Anaconda distribution. The packages can be installed in an environment, operated, and updated using Navigator. It can search for packages on Anaconda Cloud or in a local Anaconda Repository. It works with Windows, macOS, and Linux.

Conda is a package manager and environment management system that instals, runs, and updates packages and their dependencies across multiple platforms and languages. Although it was designed for Python scripts, it can package and distribute software for any language, including multi-language projects (for example, R).

All variations of Anaconda, Miniconda, and Anaconda Repository include the conda package and environment manager

C. Anaconda cloud

It is possible to locate, access, save, and share public and private notebooks, environments, conda, and PyPI packages using Anaconda Cloud, a package management service provided by Anaconda. [20] The cloud hosts practical Python environments, notebooks, and packages for a wide range of applications. For the purposes of searching for, downloading, and installing public packages, you do not need to log in or have a Cloud account. Utilizing the Anaconda Client command-line interface (CLI), you can create new packages that can subsequently be manually or automatically uploaded to the cloud.

D. Jupyter Notebook

The application can be used on a computer without an Internet connection or it can be installed on a distant server and accessed online. The dashboard and kernels are its two main parts. A kernel is a piece of software that executes and examines user code. There is a kernel for Python code in the Jupyter Notebook App, but there are other kernels for other programming languages.

You can control the kernels using the application's dashboard, which also displays the notebook pages you've created and may view again. You can see which kernels are currently running and shut them down if required.

IV. CONVOLUTIONAL NEURAL NETWORKS WORKING:

A. Convolution Of An Image

Translational invariance is a convenient characteristic of convolution. This implies that the Convolutional Neural Network algorithm learns which features make up the resultant reference, with each convolution filter representing an important feature (such as pixels in letters) (i.e. alphabet).

For a 2D image H and 2D Filter(kernel) F,

(1) Convolution Operation : $\mathbf{G} = H * F$

$$G[i,j] = \sum_{u=-k}^{k} \sum_{u=-k}^{k} H[u,v]F[i-u, j-v]$$

(2) Correlation Operation : $\mathbf{G} = H \oslash F$

$$G[i,j] = \sum_{u=-k}^{k} \sum_{u=-k}^{k} H[u,v]F[i+u, j+v]$$

Fig-2 Convolutional Neural Network Basics

We have 4 steps for convolution:

- Line up the feature and the image
- Multiply each image pixel by corresponding feature pixel
- Add the values and find the sum
- Divide the sum by the total number of pixels in the feature

B. ReLu Layer

ReLU is a function that activates.

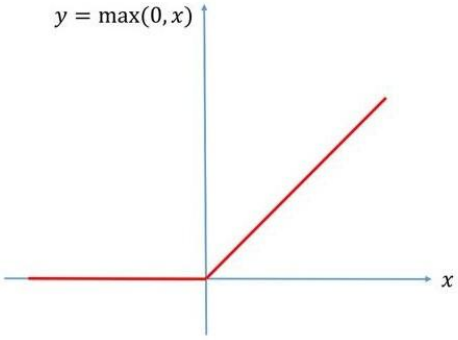

Fig-3 The ReLu Function

Rectified Linear Unit (ReLU) transform function only activates a node if the input value exceeds a predetermined threshold; otherwise, the output is zero. However, once the input value exceeds the predetermined threshold, the input has a linear relationship with the dependant variable.

C. Pooling layer

We reduce the size of the image stack in this layer. After passing through the activation layer, pooling occurs. To do this, we put the following 4 stages into practise:

- Decide on a window size (usually 2 or 3)
- Choose a stride (typically 2) and move your window
- in a circle around your filtered pictures.
- Select the highest value from each window.

V. *CONCLUSION AND FUTURE WORK*

This work used deep learning to create an automated system for detecting plant diseases. This method is built on a straightforward categorization process that makes use of CNN's feature extraction capabilities. Finally, the model uses fully connected layers for prediction. The publicly available database of photographs, taken in real-world settings and under experimental situations, was used for the research. The system functioned effectively when used to analyse photos of the plants at Sukkur IBA University and achieved a good overall testing accuracy on a publicly available dataset. From accuracy, it can be inferred that CNN is ideally suited for plant detection and diagnosis automatically. This device can be added to miniature drones to identify diseases in real time in farmed areas from plants.

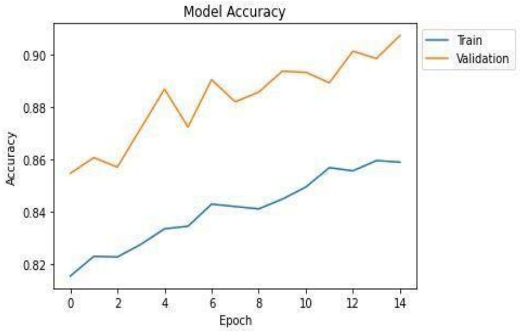

Fig-3 Accuracy graph

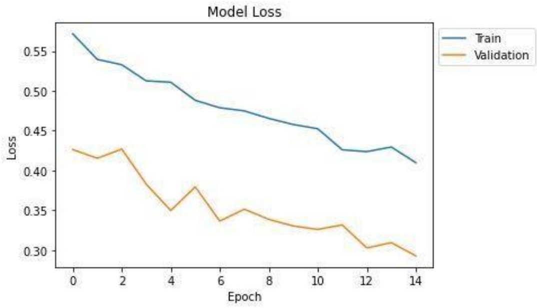

Fig-4 Loss graph

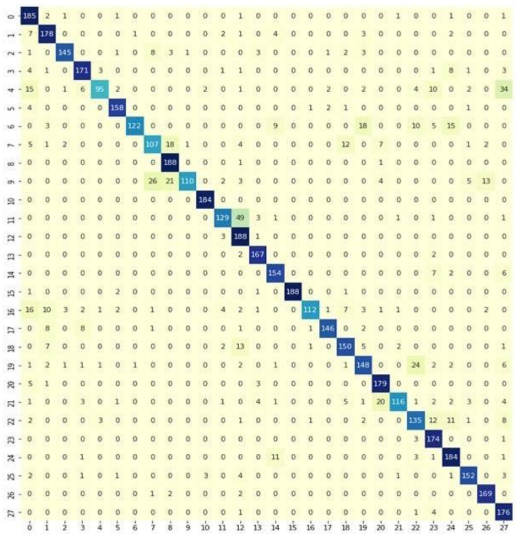

Fig-5 Experimental results

This system can detect if a plant has a disease or not even though it was trained on a small collection of classes from the Plant Village dataset. This is because all plants exhibit some symptoms that are similar. Additionally, new photographs of the actual environment can be added to the dataset to increase classification accuracy for real-world leaf images and categorise more plant and disease species. In the future, this system may also use a three-layer approach, where the first layer determines whether or not an image contains any plants, the second layer identifies the type of plants, and the third layer determines whether or not there are any diseases present and, if so, what kind of diseases they are.

REFERENCES

[1] Dhiman Mondal; Aruna Chakraborty; Dipak Kumar Kole; D. Dutta Majumder,“Detection and Classification technique Of Yellow Vein Mosaic Virus Disease in Okra Leaf Images Using Naive Bayesian Classifier”, International Conference on Soft Computing Techniques and Implementations (ICSCTI) 13 June 2016

[2] Pranjali B.Padol; Anjali A. Yadav, “Svm Classifier Based Grape Leaf Disease Detection”, Conference on Advances in Signal Processing (CASP) 17 November2016

[3] Zarreen Naowal Reza; Faiza Nuzhat; Nuzhat Ashraf Mahsa; Md. Haider Ali, “Detecting Jute Plant Disease Using Image Processing and Machine”, International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), 24 September 2016

[4] M S Arya; K Anjali; Divya Unni, “Detection Of Unhealthy Plant Leaves Using Image Processing and Genetic Algorithm with Arduino”, International Conference on Power, Signals, Control and Computation (EPSCICON), 14 June 2018

[5] Tanvi Mehra; Vinay Kumar; Pragya Gupta, “Maturity and disease detection in tomato using computer vision” Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC), 27 April 2017

[6] Pooja Pawar; Varsha Turkar; Pravin Patil ,“Cucumber disease detection using artificial neural network.” International Conference on Inventive Computation Technologies (ICICT),26 January 2017

[7] R. P. Narmadha; G. Arulvadivu, “Detection and measurement of paddy leaf disease symptoms using image processing International Conference on Computer Communication and Informatics (ICCCI), 23 November 2017

[8] Mukesh Kumar Tripathi; Dhananjay D. Maktedar. “Recent Machine Learning Based Approaches for Disease Detection and Classification of Agricultural products.”, International Conference on Computing Communication Control and automation (ICCUBEA), 23 February 2017