# Identify the category of Foliar disease

Pratiksha Wable[1], Paritosh Jain[2], and Prof. Pankaj Chandre[3]
[1,2]*Student, Computer Science and Engineering MIT School of Engineering, MIT ADT University, Pune*
[3]*Associate Professor, Computer Science & Engg MIT School of Engineering, MIT ADT University, Pune*

**Abstract -Accurate identification and early diagnosis of apple tree leaf disease can control the spread of infection, to reduce use of chemical fertilizers and pesticides, to improve the quality of apples and maintain the healthy development of apples. First, collect the images of leaves with or without disease from Kaggle, and it contains two common apple tree leaf diseases and healthy leaves. The common leaf diseases are scab, multiple diseases and Rust. We build our model using convolution neural network. The proposed network achieves overall accuracy of 96% in identifying the apple tree leaf disease..**

***Index Terms*—CNN, Machine learning, keras, Tensorflow**

## I. INTRODUCTION

It is a big challenge for agricultural products to feed the world's growing population. So sustainability of farming products against diseases is a very important part of increasing the yield of the products. Plants and trees face biotic stresses and adverse environmental and weather conditions. To deal with these challenges, farmers use different techniques, including pesticides, fertilizers, irrigation policies, etc. Fruit agriculture help to provide food for human beings and the enjoyment of different tastes. Apple is a very important tree because of its fruit, which is one of the most cultivated and consumed fruit in the world, because of its high nutritional and medicinal values. On an average, over 10,000,000 tonnes of apples are being added and produced every year. However, diseases in apple leaves cause major production and economic losses and reductions in both the quality and quantity of the fruit industry output. Only with proper care and control of insects and proper usage of fertilizers we can save trees. Therefore, initial detection and correct classification of disease help the farmer take action in time.

The current diagnosis and manual examining process are moderate and any misdiagnosis of the disease leads to false use of chemicals and fertilizers, which leads to the evolution and development of new resistant bacteria, making the situation worse. Machine learning-based detection and recognition methods of apple diseases have been proven to be beneficial in monitoring large apple fields, which provide clues to treat the apple diseases at their early stages. To control apple diseases, the first step is to detect the disease and identify the type of disease. The leaves of the apple tree are abundant, occupy most parts of the tree, and are the easiest part to observe. Considering that a lot of apple diseases can be recognized through their diseased leaf symptoms, our research interest is in the apple leaves rather than the whole apple tree. Convolutional Neural Networks are perfect for image processing tasks. The convolutional layer is the building block of a convolutional neural network, which comprises several independent filters and each filter is independently convolved with images and performing dot products of previous convolutional layer input, similar to the feedback of neurons in the brain for a specific stimulus

## II. DATASET

To complete the identification of foliar disease in apple trees, first step is to collect the dataset. The data used is extracted from Kaggel PLANT PATHOLOGY 2020 FGVC7. The dataset consists of 3641 images of Apple tree leaves categorized into 4 different fields healthy, rust, scab and multiple disease. Then we complete the dataset preprocessing tasks such as data normalization, dataset expansion and image scaling.. Finally the dataset is divided into two parts such as training, testing.


Healthy


Rust


Multiple disease


Scab

*Dividing dataset*

For training and testing of CNN, dataset with 1820 images was divided into two independent subsets, including training dataset and testing dataset 1821. The training dataset used to train the model.Dataset Image preprocessing:

We apply data augmentation and data preprocessing on images.

Data Augmentation:

Convolutional neural networks are their ability to generalize, their ability to process data that has never been observed. when data size is not large enough and the data diversity is limited, then tend to overfit the training-data which means they cannot be generalized. In order to enhance generalization ability of the network and reduce overfitting, the dataset was expanded by data augmentation technology to simulate changes in lightning, exposure, angle during preprocessing step of apple leaf image. Table shows the number of images in the training dataset.

| Disease Type | Original Images |
|---|---|
| Rust | 622 |
| Scab | 591 |
| Multiple disease | 91 |
| Healthy Leaves | 516 |

## III. CONSTRUCTING NEURAL NETWORK

TensorFlow is a deep learning framework developed by google. Using TensorFlow directly can be challenging, the modern tf.keras API brings the simplicity and ease of use of Keras to the TensorFlow project.

Keras is deep learning library written in python. The focus of these libraries was on flexibility ,research and speed. Keras is popular because the API was clean and simple, allowing standard deep learning models to be defined, fit and evaluated in just a few lines of code. The secondary reason keras took-off was because it allowed you to use any one among the range of popular deep learning mathematical libraries as the backend(e,g used to perform the computation) such as TensorFlow.

Using tf.keras allows you to design ,fit, evaluate, and use deep learning models to make predictions in just a few lines of code. It makes common deep learning tasks, such as classification and regression predictive modeling, accessible to average developers looking to get things done.

The model has a life-cycle, and this very simple knowledge provides the backbone for both modeling a dataset and understanding the tf.keras API

The five steps in the life-cycle are as follows:

1. Define the model
2. Compile the model
3. Fit the model
4. Evaluate the model
5. Make predictions

1.Define the model: Defining a model requires that first select the type of model that is needed. In this Project the sequential model is used. From API perspective, this involves defining the layers of the model, configuring each layer with a number of nodes and activation function, and connecting the layers together into a cohesive model

2.Compile the model: Compiling the model requires that first select a loss function that you want to optimize, such as mean squared error or cross-entropy. It also requires that you select an algorithm to perform the optimization procedure, typically stochastic gradient descent or modern variation, such as Adam. From an API perspective, this involves calling a function to compile the model with the chosen configuration, which will prepare the appropriate data structure required for the efficient use of the model you have defined.

a.Categorical cross-entropy: Categorical cross-entropy is a loss function that is used in multi-class classification tasks. categorical cross- entropy is used

when we want to one output from many possible categories, and the model must decide which output.

b.Loss function: A loss function is used to measure how good is prediction model in terms of predict the right outcome. We convert the learning problem into an optimization problem, define a loss function and the optimize the algorithm to minimize the loss function

c.Softmax: Softmax: Softmax is continuously differentiable function. This makes it possible to calculate the derivative of the loss function with respect to every weight in the neural network. This property allows the model to adjust the weight accordingly to minimize the loss function (model output close to the true values)

d.Optimizer: An optimizer is function which is used to modifies the attributes of neural network, such as weight and learning rate. An Optimizer helps to reduce loss and improve accuracy

3.Fit the model: Fitting the model requires that first select the training configuration, such as number of epochsand the batch size(number of samples in an epoch used to estimate model error)

Fitting the model is the slow part of the whole process and can take seconds to hours to days, depending on the complexity of the model, the hardware you are using, and the size of the training dataset. From an API perspective, it involves calling a function to perform the training process. This function will block until the training process has finished

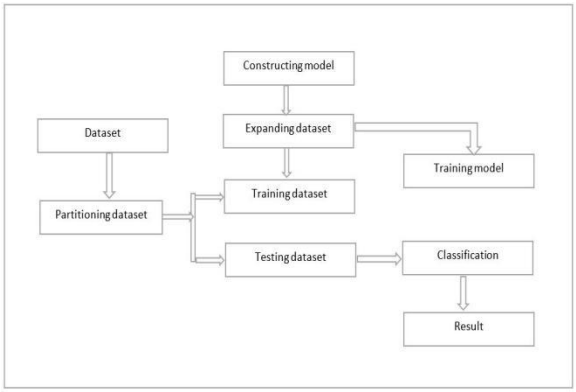
4. Evaluate the model: Evaluating the model requires that first choose the dataset used to evaluate the model. The speed of model evaluation is proportional to the amount of data we want to use for the evaluation, as it is much faster than training as the model is not changed.

From API perspective, this involves calling a function with the dataset and getting a loss and perhaps other metrics that can be reported

5. Make predictions: Making a prediction is the final step in the life-cycle. That is why we needed the model in the first place. It requires you to have new data for which a prediction is required, e.g where you do not have the target values

From an API perspective, you simply call a function to make a prediction of class label, probability or numerical value. Whatever you designed your model to predict. You may want to save the model and later load it to make predictions. You may also choose to fit a model on all of the available data before you start using it.

## IV. DETECTION PROCESS




Apple tree leaf diseases detection process is shown in the figure below. We collect the dataset of diseased leaves and healthy leaves of apple trees from Kaggle. The original dataset is divided into categorical apple disease of apple. After that, we perform the data augmentation on the training dataset and all images are normalized. Finally, the specific disease type of each image in the testing dataset was detected by the model.

## V.CONFIGURATION

Project is implemented in keras based on CNN using python language. The configuration parameters are below:

| Configuration Item | value |
|---|---|
| CPU | Intel(R) Core(TM) i3-7100U CPU @ 2.40GHz 2.40 GHz |
| RAM | 8.00 GB (7.87 GB usable) |
| Hard Disk | 1TB |
| Operating System | Windows 10 |

## VII. CONCLUSION

Agriculture is considered an engine for developing economies and industrialized countries. It is source of food consumed by human beings and animals. Plant disease causes considerable losses to the agriculture sector, which results in less production. In this study, a convolution neural network was trained to detect diseases in plant leaves using transfer learning

approaches. Early diagnosis and detection might control the spread of the disease in the early stages to mitigate the losses resulting from the disease.

Prior detection and identification of diseases in trees help to eradicate the misuse of chemical fertilizers and improve production. This paper proposes the usage of Deep Ensemble Neural Networks and identification of foliar disease in Apple trees by a single picture of the Apple tree leaves.

REFERENCE

[1] https://www.researchgate.net/publication/352247462_Identification_of_Apple_Tree_Leaf_Diseases_Based_on_Deep_Learning_Models
[2] https://medium.datadriveninvestor.com/detection-and-classification-of-apple-tree-disease-based-on-deep-learning-algorithm-d8c29940ed02
[3] https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html
[4] https://www.academia.edu/44230991/IRJET_Automated_Diagnosis_and_Cataloguing_of_Foliar_Disease_in_Apple_Trees_using_Ensemble_of_Deep_Neural_Networks