HOCHSCHULE
RAVENSBURG-WEINGARTEN
UNIVERSITY
OF APPLIED SCIENCES

Embedded and Electrical Engineering

Hochschule Ravensburg-Weingarten University

Deep Learning

Shubham Kadamdhad
ID: 32920

Under guidance of Prof. Mark Schutera

1. Abstract:

Immense commercialisation of an agriculture has creates a very negative effect on our environment. The use of chemical pesticides has led to enormous levels of chemical build-up in our environment, in soil, water, air, in animals and even in our own bodies. Artificial fertilizers gives on a short-term effect on productivity but a longer-term negative effect on the environment, where they remain for years after leaching and running off, contaminating ground water. Another negative effect of this trend has been on the fortunes of the farming communities worldwide. Despite this so-called increased productivity, farmers in practically every country around the world have seen a downturn in their fortunes. This is where organic farming comes in. Organic farming has the capability to take care of each of these problems. The central activity of organic farming relies on fertilization, pest and disease control. Using CNN and Machine Learning farmers can easily identify the diseases plants are having can make a Important decision as there needs to protect the farms and vegetables.[1]

2. Introduction:

Here we will evaluate aspects like description, problem statement and also the objectives of the project.

1) **Description** - The state of the art method of identifying diseases in plants in the field is use of visual symptoms which an agricultural expert is able to relate to particular diseases in the plant. For places where experts are not available or where farmer knowledge is insufficient, other methods for carrying out field-based diagnoses are a critical need. Computational work in this area has been towards automating this process through building machine learning models that can take an image of a leaf and predict whether the plant is infected with a particular disease or not.

2) **Problem Statement -** Misdiagnosis of the many diseases impacting agricultural crops can lead to misuse of chemicals leading to the emergence of resistant pathogen strains, increased input costs, and more outbreaks with significant economic loss and environmental impacts. Current disease diagnosis based on human scouting is time-consuming and expensive, and although computer-vision based models have the promise to increase efficiency, the great variance in symptoms due to age of infected tissues, genetic variations, and light conditions within trees decreases the accuracy of detection.[3]

3) **Specific Objectives -**To train a model using images of training dataset to
    I. Accurately classify a given image from testing dataset into different diseased category or a healthy leaf;
    II. Accurately distinguish between many diseases, sometimes more than one on a single leaf.
    III. Deal with rare classes and novel symptoms.
    IV. Address depth perception—angle, light, shade, physiological age of the leaf.
    V. Incorporate expert knowledge in identification, annotation, quantification, and guiding computer vision to search for relevant features during learning. [3]

3. Dataset :

For this project I have used a dataset which is already given in Kaggle competition. The folder structure of the dataset is given below.

- Images(.jpg)
- Train(.csv)
- Test(.csv)
- sample_submission(.csv)

The set of images which are to be trained and tested are given in the images folder. The dataset contains the total of 1821 training images and 1821 testing images. The Train.csv contains all the training parameters of the image. The parameters are those which are healthy, those which are infected with apple rust, those that have apple scab, and those with more than one disease. The Test.csv contaains all the ids of testing images for which the model would be evaluated. Kaggle has not published any labels of the testing images of the dataset. The fig 1. shows some of the sample images from the dataset.

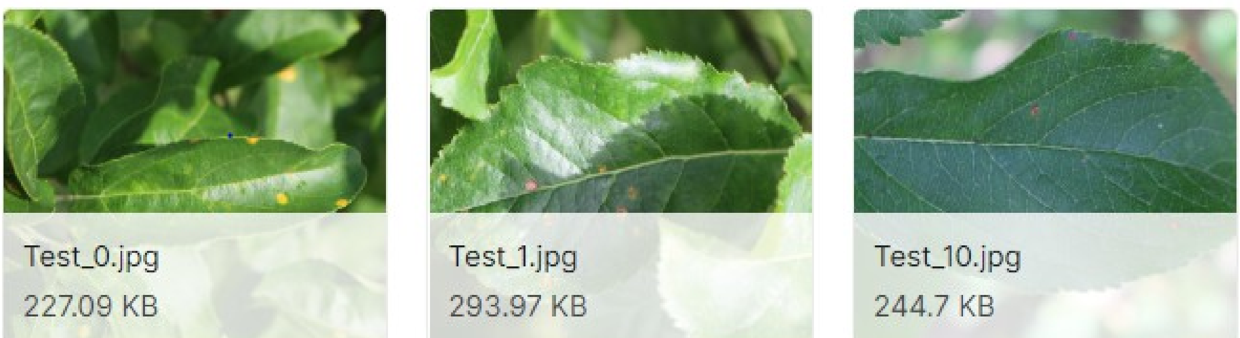


Fig 1.[4]

4 .Detection and classification of plant Disease using CNN in Keras

In this paper,I have used a Convolutional Neural Network (CNN) architecture to classify the type of plants steps. Following the pre processing step, Convolutional Neural Network architecture is employed to extract the features of images. I have used a reference of already pretrained model on Kaggle and necessarily done modification of the parameters of the model according to the requirement. The API used is Keras.[5]

Proposed System :

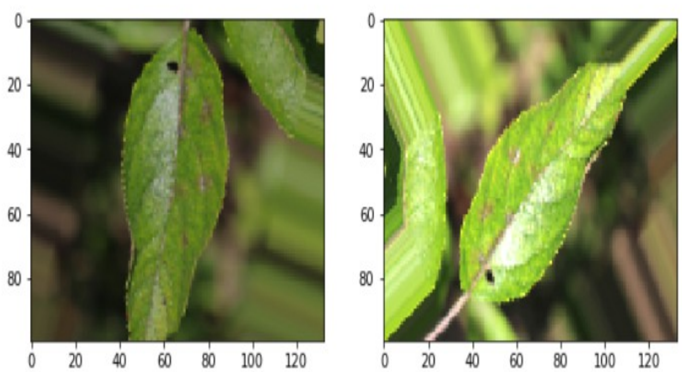
The project is basically done in four steps:
1)Image preprocessing
2)Building of convolution neural network
3)Training of model
4)Measuring accuracy of the model using test images.

1)Image Preprocessing :

For our neural network to work better we have done preprocessing of the images. For our neural network to work better with the set of different images we have done data augmentation using the ImageDataGenerator(Class in python for Image Augmentation).
Applying moderate amount of zoom in/out and brightness variation. Full rotation and flips are applied since ther e is no obvious orientation that the pictures of the leafs are taken. We have Defined the ImageDataGenerator using a Training and Validation split of 80% and 20% respectively.
The Parameters of Data Augmentation which we used are given below :

```
shear_range=0,
zoom_range=(1, 1.3),
rotation_range = 360,
brightness_range = (0.7, 1.3),
horizontal_flip=True,
vertical_flip=True,
validation_split=0.2)
```

We have validated all the Training and testing images using the above Data Augmentation parameters.



The result of applying augmentation on the same picture twice is shown in the Figure 2.

2)Building a convolutional neural network:

In deep learning, a convolutional neural network(CNN) is a class of deep neural networks, most commonly applied to analyzing visual imagery.[2]

CNNs use a variation of multilayer perceptron's designed to require minimal preprocessing. They are also known asshift invariant or space invariant artificial neural networks (SIANN), based on their shared-weights architecture and translation invariance characteristics.[2]

Convolutional networks were inspired by biological processes in that the connectivity pattern between neurons resembles the organization of the animal visual cortex. Individual cortical neurons respond to stimuli only in a restricted region of the visual field known as the receptive field. The receptive fields of different neurons partially overlap such that they cover the entire visual field.[2]

CNNs use relatively little pre-processing compared to other image classification algorithms. This means that the network learns the filters that in traditional algorithms were hand-engineered. This independence from prior knowledge and human effort in feature design is a major advantage.[2]

Hyperparameters	Model(CNN)
Batch Size	32
Epoch	50
Activation	RELU
Optimizer	ADAM
Learning Rate	o.oo125

Fig 3. The parameters used for training of model

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 100, 133, 35)      980
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 100, 133, 35)      11060
_________________________________________________________________
dropout_1 (Dropout)          (None, 100, 133, 35)      0
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 50, 66, 35)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 48, 64, 35)        11060
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 46, 62, 35)        11060
_________________________________________________________________
dropout_2 (Dropout)          (None, 46, 62, 35)        0
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 9, 12, 35)         0
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 7, 10, 50)         15800
_________________________________________________________________
conv2d_6 (Conv2D)            (None, 5, 8, 50)          22550
_________________________________________________________________
dropout_3 (Dropout)          (None, 5, 8, 50)          0
_________________________________________________________________
global_max_pooling2d_1 (Glob (None, 50)                0
_________________________________________________________________
dropout_4 (Dropout)          (None, 50)                0
_________________________________________________________________
```

Fig4. The Network architecture of CNN Model.

3)Training of Model :

As shown in the Fig 3 we have used all those parameters to train the network with the kernal size of 3*3. The network used cross entropy loss as classification loss. The loss is given by below equation.

$$l_{cls} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{c_j}^{i} \log\left(p_{c_j}^{i}\right)$$

Here y and p are class label and class probability of the i'th vertex respectively.

We have initially split the dataset into 80% and 20% train and validation set and later we have trained our model with the set of all the images. The Accuracy and Loss plot against Epochs is shown in figure 5. and figure 6. respectievely.

4)Testing the model :

Initially we have split dataset in 80/20 train and validation set, since we have limited labeled data the accuracy of the model with 80% of testing images was around 0.87. Since the labeled data is limited we observe that overfitting is not an issue so we proceed to train the model over all the images to increase the accuracy of the model for the unlabeled data, and the result was positive. The final accuracy we achieved with the testing images is 0.94.

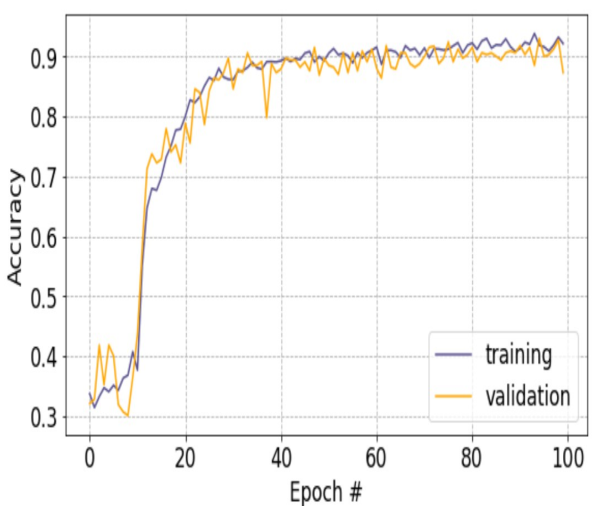


Fig 5. Epoch vs Accuracy plot for 100 Epochs

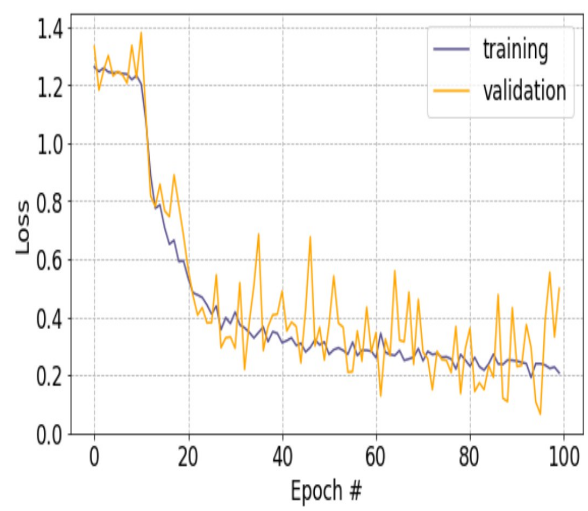


Fig 6. Epoch vs Loss plot for 100 Epochs

Result :

Thus we finally submitted our result to kaggle. And our model shows the accuracy of around 94% for the set 1821 testing images.

Conclusion :

Crop protection in organic agriculture is not a simple matter. It depends on a thorough knowledge of the crops grown and their likely pests, pathogens and weeds. In our system specialized deep learning models were developed, based on specific convolutional neural networks architectures, for the detection of apple plant diseases through leaves images of healthy or diseased apple plant images. Our experimental results an demonstrated how our deep-learning-based detector is able to successfully recognize different categories of diseases in apple plant leaves.

References :

[1]. Aakanksha Rastogi, Ritika Arora, Shanu Sharma, "Leaf Disease Detection and Grading using Computer Vision Technology &Fuzzy Logic," presented at the 2nd International Conference on Signal Processing and Integrated Networks (SPIN), IEEE, 2015, pp. 500–505.

[2]. Garima Tripathi, Jagruti Save, "AN IMAGE PROCESSING AND NEURAL NETWORK BASED APPROACH FOR DETECTION AND CLASSIFICATION OF PLANT LEAF DISEASES," Int. J. Comput. Eng. Technol. IJCET, vol. 6, no. 4, pp. 14–20, Apr. 2015.

[3]. https://www.kaggle.com/c/plant-pathology-2020-fgvc7/overview

[4]. https://arxiv.org/abs/2004.11958

[5]. https://www.kaggle.com/c/plant-pathology-2020-fgvc7/notebooks