Deep CNN based Detection System for Real-time Plant Disease Recognition

Aashay Tiwari

Department of Computer Science and Engineering

SRM Institute of Science and Technology

Raghav Shankar

Department of Computer Science and Engineering

SRM Institute of Science and Technology

Prof. M Vaidhehi

Department of Computer Science and Engineering

SRM Institute of Science and Technology

Abstract

Profitability in horticulture is profoundly monetary ward. This is one reason why the finding in agribusiness of plant infections is significant on the grounds that plant illness is exceptionally normal. Food security for the 7 billion individuals on earth requires limiting yield harm by ideal identification of infections. In the event that legitimate consideration isn't taken around there, the effect on plants and the quality, amount, or profitability of their items will be not kidding.

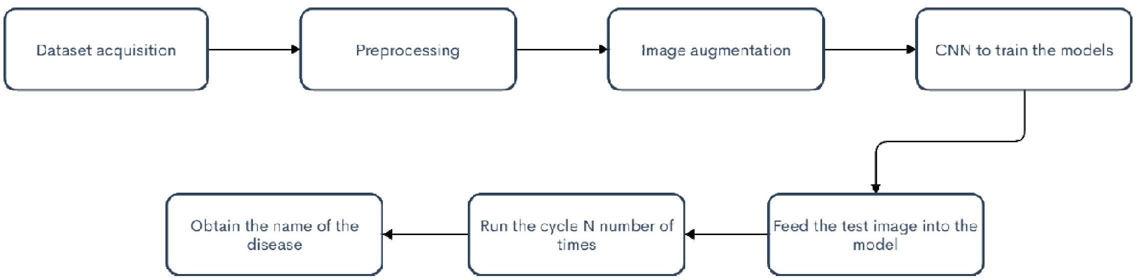
It is useful to determine plant illness to have any robotized methodology, as it diminishes the inescapable observation of ranch locales, and recognizes the indications of infections early, that is the point at which they emerge on leaves of the plant. This paper presents a calculation for image segmentation strategy which is utilized for programmed location and arrangement of plant leaf infections.

image segmentation, which is a significant angle for sickness discovery in plant leaf illness, is finished by utilizing a genetic algorithm. This work explores a possible answer for this issue by utilizing portioned picture information to prepare the convolutional neural organization (CNN) models.

Introduction

Plant disease detection is very important in rural areas where this a lot of farming. This can be useful to the farmers in such a way that they will know how to treat the disease. This model will be useful in that way. It can also be used to decide whether or not this plant can be consumed.

Our project entails the quality of fruits and vegetables. But this time we are doing this with the help of leaves of the plants by detecting what diseases they have, and whether or not we can eat them. The dataset we have taken is from plantvillage, where different fruits and vegetable leaves are stored. They have photos of healthy and affected leaves, so that the model can differentiate between them. If the leaf is healthy, we can eat that food item, and if it is not, we cannot.



Flowchart of the model

Starting off with the dataset, we have chosen one with 1000s of photograph, so that the model can be trained accurately. It is split in to 80% training and 20% test data, so the photos that are used to test the model later will be randomized. Then coming to preprocessing, the photos are converted to arrays for faster training of the model. Image augmentation is done for increasing the size of the dataset. It is rotated, zoomed in, and horizontally flipped to create more images so that the model is accurate with its prediction.

We are going to use Convolutional neural networks that helps us in training the model.CNN has input layers, hidden layers and output layers. We feed in the information in the input layers, it processes the image in the pooling layers, the fully connected layers and the normalization layers, and it gives an output. This is done n number of times for giving a more accurate score.

After the model is trained, it is stored as a file, and we test the model with different images from the dataset. The images we supply it are first converted into an array format, and then fed into the model. It gives the output in an array format. That when converted gives us the result of the disease.

Literature Survey

In order, to explore the research work which has been done in the past on this topic, we referred to some of the studies dedicated to the same. All the literature survey paper listed below are quite recent and belong from the year 2010 to 2020.

Sumita Mishra in 2019 developed a deep convolution neural network-based detection system to recognize any real time corn plant disease. She used a pre trained CNN and was deployed over raspberry pie 3. While recognizing the diseases present in corn plant, the model obtained an accuracy of 88.46%

Rajesh Kumar Mojjada in the year 2020, developed a CVS and trained that computer system to predict any disease present on the surface of plant's leaves by using image segmentation and digital image processing.

Parul Sharma in 2019, developed a deep CNN model and trained it to study 15 different types of plant species. They captured an image of the plant of size 256*256 and after preprocessing the image, then a convolution filter was applied all over the image inorder to train the CNN model. This paper involved use of both F-CNN and S-CNN. The accuracy obtained by them was 93.6%.

Vijai Singh in 2016 developed a CVS to detect plant diseases using image segmentation and soft computing techniques. They used a technique called genetic algorithm for image segmentation. The accuracy obtained by this was 86.4%.

Mrunalini R et al in 2018 presented the method to characterize and distinguish the diverse infection through which plants are influenced. In Indian Economy a Machine learning based acknowledgment framework will end up being exceptionally valuable as it saves endeavors, cash and time as well. The methodology given in this for highlight set extraction is the Color Co-event Method. For programmed recognition of illnesses in leaves, neural organizations are utilized. The methodology proposed can essentially uphold a precise location of leaf, and is by all account's significant methodology, if there should arise an occurrence of steam, and root infections, investing less amounts of energy in calculation.

Savita N. Ghaiwat et al. in 2016 presented overview on various classification procedures that can be utilized for plant leaf sickness characterization. For given test model, k-nearest neighbor technique is by all accounts reasonable just as least complex of all calculations for class forecast. In the event that preparation information isn't directly distinguishable, it is hard to decide ideal boundaries in SVM, which shows up as one of its downsides.

Anand H. Kulkarni et al. in 2019 introduced a technique for exactly on schedule and unequivocally plant sicknesses revelation, using artificial neural network (ANN) and diverse picture handling systems. As the proposed approach relies upon ANN classifier for plan and Gabor channel for feature extraction, it gives better results with an affirmation speed of up to 91%. An ANN based classifier orders unmistakable plant contaminations and uses the mix of surfaces, concealing and features to see those afflictions.

Dataset for corn leaf disease classification.

Class	Common Rust	Northern Leaf Blight	Healthy
No.of Train Images	1192	986	1162
No.of Test Images	139	100	124
No.of Validation Images	227	291	161

Proposed Work

The proposed system makes use of Convolutional neural networks which is then applied on the dataset acquired from plantvillage. The use of CNN and image analysis in this system makes it a method of obtaining foolproof empirical results. We can lessen the assault of vermin by utilizing legitimate pesticides and cures.We can decrease the size of the pictures by

appropriate size decrease methods and make sure that the quality isn't undermined by and large. We can extend the tasks of the prior referenced creators to such an extent that the solution for the illness is likewise appeared by the framework. The principal objective is to recognize the plant illnesses utilizing picture preparing. It likewise, after distinguishing proof of the sickness, propose the name of pesticide to be utilized. It likewise recognizes the creepy crawlies and bugs answerable for plague. Aside from these equal targets, this robot is very efficient. The financial plan of the model is very high for low scope cultivating purposes yet will be an incentive for cash in enormous scope cultivating. It finishes every one of the interactions consecutively and henceforth accomplishing every one of the yields.

Accordingly the primary destinations are:

1) To plan such framework that can identify crop infection and irritation precisely.

2) Create data set of bug sprays for separate irritation and illness.

3) To give solution for the infection that is distinguished.

Preprocessing

First, we make a function to convert images to arrays so that all the images are of the same format and they are understood by the model. Then the dataset is accessed to acquire all the images. The program checks for all the images that end with .jpg or .JPG so that all images are of same format. These images are passed through the function one by one so they are all converted.

Classification and Augmentation

We use the size of each image and the label binarizer to classify different leaves of fruits and vegetables. The arrays are then converted to a smaller float value for the model to be trained properly. Coming to augmentation, it is necessary as it increases the accuracy of the model. It is done to the dataset to increase the number of images in it by rotating, zooming in, flipping it horizontally and vertically, etc. All this data is then fed into our Convolutional neural network which is then used to make the model.

```python
# %% [markdown]
#build our model

model = Sequential()
inputShape = (height, width, depth)
chanDim = -1
if K.image_data_format() == "channels_first":
    inputShape = (depth, height, width)
    chanDim = 1
model.add(Conv2D(32, (3, 3), padding="same",input_shape=inputShape))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(MaxPooling2D(pool_size=(3, 3)))
model.add(Dropout(0.25))
model.add(Conv2D(64, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(Conv2D(64, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Conv2D(128, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(Conv2D(128, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(1024))
model.add(Activation("relu"))
model.add(BatchNormalization())
model.add(Dropout(0.5))
model.add(Dense(n_classes))
model.add(Activation("softmax"))
```

CNN based model

Related Work

Image Segmentation

Image segmentation is the path toward disconnecting or assembling an image into different parts. There are as of now different techniques for performing Image segmentation, going from the essential thresholding system to front line concealing Image segmentation procedures. These parts commonly identify with something that individuals can without a doubt separate and view as individual articles. Laptops have no techniques for shrewdly seeing items, along these lines different systems have been made to portion pictures.
The division interaction depends on different highlights found in the picture. This may be shading data, limits, or section of a picture. We utilize a Genetic algorithm for shading Image segmentation.
Developmental processing was first presented during the 1960s by I. Rechenberg. His thought was then taken forward by different scientists. In some cases, transformative changes are little and seem unimportant from the start, however they have an impact in normal determination and the endurance of the species.

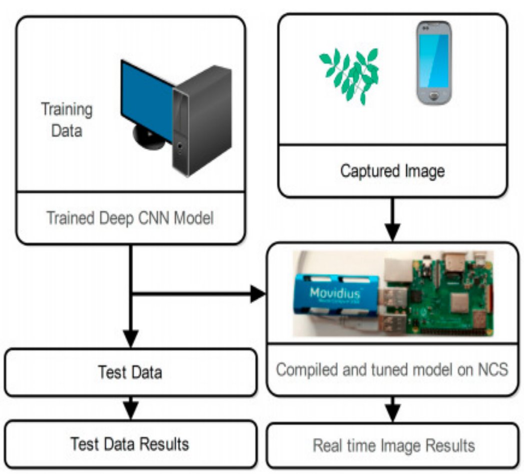
F-CNN

Huge earlier work in the pertinent zone incorporates field image-based profound neural network by De Chant et al for northern leaf curse identification. Creators have prepared a few CNNs to group little locales of pictures into tainted and sound classes and afterward, these little districts are taken care of into the last CNN that orders the entire picture as unhealthy or sound. The proposed framework accomplished high exactness on test information. A few specialists applied exchange learning on prior models prepared on various information to improve the grouping exactness of plant illnesses. These examinations have gotten essential outcomes, however, to acquire a doable answer for exactness crop observing it is profoundly alluring to configuration plant infection distinguishing proof techniques that can give sensible precision on independent cell phones without the prerequisite of Internet access. It will empower ranchers to settle on brisk and precise choices about crop illness.

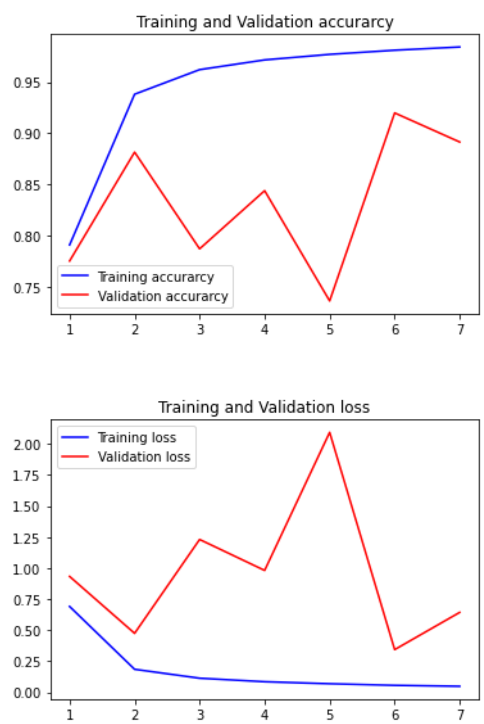
Implementation

Convolutional Neural Network

The process starts with the creation of Convolutional neural networks in which we are supplying images of leaves which are affected by diseases. The network is created by taking several images, augmenting them by various methods and then feeding the dataset to the model for training and testing purposes.



CNN Example



Accuracy graphs

The dataset contains images of leaves of different fruits, vegetables and other food items which are either healthy or are affected by diseases. To get the required database we augment all the images by doing certain operations on them. This ensures to get a variety of images which will increase the accuracy of the model.The database is labelled and it is split into training and testing data. 80% is the training data, and 20% is the testing data. Also, all the images that will be trained are randomly done, so while testing the model you would not know which photo was trained and which was not. Several libraries such as NumPy, cv2, Keras, etc. are imported and TensorFlow backend is used for yielding the output.

The process we are undergoing starts with converting the images to arrays. Each image is divided into red, blue and green pixels of arrays. A sequential model is used to map an input to the output. These images are converted into layers which are fed into the model which then gets trained and gives an output. These layers are normalised using a layer called batch normalisation, which standardises the output of the previous layers. The activations scale the input layer in normalisation.

```
Epoch 1/7
6041/6041 [==============================] - 1598s 265ms/step - loss: 0.6748 - accuracy: 0.7
957 - val_loss: 0.7859 - val_accuracy: 0.7962
Epoch 2/7
6041/6041 [==============================] - 1637s 271ms/step - loss: 0.1827 - accuracy: 0.9
387 - val_loss: 1.5376 - val_accuracy: 0.7141
Epoch 3/7
6041/6041 [==============================] - 1627s 269ms/step - loss: 0.1117 - accuracy: 0.9
634 - val_loss: 0.7692 - val_accuracy: 0.8253
Epoch 4/7
6041/6041 [==============================] - 1643s 272ms/step - loss: 0.0809 - accuracy: 0.9
732 - val_loss: 0.5459 - val_accuracy: 0.8809
Epoch 5/7
6041/6041 [==============================] - 1617s 268ms/step - loss: 0.0632 - accuracy: 0.9
790 - val_loss: 0.3481 - val_accuracy: 0.9133
Epoch 6/7
6041/6041 [==============================] - 1634s 271ms/step - loss: 0.0545 - accuracy: 0.9
821 - val_loss: 0.3185 - val_accuracy: 0.9305
Epoch 7/7
6041/6041 [==============================] - 1637s 271ms/step - loss: 0.0482 - accuracy: 0.9
843 - val_loss: 0.6086 - val_accuracy: 0.8928
```

Model Compilation process

Results discussion

The photos used for plant disease detection are first preprocessed into arrays and then augmented into many other images. After this, the images are classified into many diseases and put in different folders. This is then sent to the model for training. We then test the model with an image from the database itself. It then gives us an accuracy of 89%. This is better than the accuracy of digital image processing and image segmentation. The fact that we did not use image segmentation saved a lot of time and also increased the accuracy.

In [20]:

```
from IPython.display import Image
Image('../input/plantvillage/PlantVillage/Tomato/Tomato___Septoria_leaf_spot/0025c401-7785-49c5-8bef-780a8a0d3652___Matt.S_CG 2694.JPG')
```

Out[20]:

In [23]:

```
#TESTING THE MODEL HERE
res = model.predict(img)

res
```

Out[23]:

```
array([[2.57761174e-12, 3.65191800e-11, 3.60247162e-12, 3.60295750e-13,
        2.60067540e-11, 3.31083969e-14, 1.00190889e-09, 1.36272707e-13,
        8.50600233e-12, 5.34633782e-10, 4.10518064e-10, 4.56676190e-07,
        3.08441085e-11, 1.40826089e-07, 1.10006482e-09, 1.05222792e-11,
        1.08513025e-11, 7.57743868e-12, 3.66501886e-13, 3.36591683e-14,
        9.04647468e-09, 2.06099325e-13, 4.29649480e-13, 4.93964369e-10,
        1.06231790e-09, 2.71477330e-09, 4.03250766e-08, 4.28664837e-09,
        2.53010949e-05, 7.70675954e-11, 3.32522510e-10, 1.24567052e-07,
        9.99973774e-01, 1.55163712e-11, 1.22004362e-09, 4.04269979e-10,
        4.79618185e-11, 1.59232385e-07]], dtype=float32)
```

In [24]:

```
#IMPORT LABEL_TRANSFORM.PKL HERE!

print(label_binarizer.inverse_transform(res))
```

```
['Tomato___Septoria_leaf_spot']
```

For testing the model, we copy the path of an image from the database and we supply it to the model. It is first converted to an array and then fed into the model. The model gives the output

in an array format. Then we use the array to string function as shown above to convert the array to string. That gives us the required output. It is shown in the previous figure. Since the leaf of the plant is infected there is a high probability that the fruit or vegetable of the same plant is infected. This renders the food inedible. If eaten, it can cause major health issues and sometimes can be hazardous.

Conclusion

The methods above can be used for detection of diseases in plants which will eventually lead us to the fact whether that fruit or vegetable can be eaten or not. Such diseases might not be noticeable to the naked eye, but the model which we have made will do it for us and will do it better than models created using Image segmentation or digital image processing. The applications are endless. It can be used on farms for detecting any diseases, or it can be used in a garden where someone might grow a plant. This model is based off a small dataset, this can be used as a base model for other researchers to develop on it to perfect the system and increase the accuracy of detection.

References

- [1] Savita N. Ghaiwat, Parul Arora. Detection and Classification of Plant Leaf Diseases Using Image processing Techniques: A Review. International Journal of Recent Advances in Engineering & Technology. ISSN (Online): 2347 - 2812, Volume-2, Issue - 3, 2014
- [2] Prof. Sanjay B. Dhaygude, Mr.NitinP.Kumbhar. Agricultural plant Leaf Disease Detection Using Image Processing. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering. Vol. 2, Issue 1, January 2013
- [3] Mrunalini R. Badnakhe and Prashant R. Deshmukh. An Application of K-Means Clustering and Artificial Intelligence in Pattern Recognition for Crop Diseases. International Conference on Advancements in Information Technology. 2011 IPCSIT vol.20 (2011)
- [4] S. Arivazhagan, R. Newlin Shebiah, S. Ananthi, S. Vishnu Varthini. Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture features. Agric Eng Int: CIGR Journal, 15(1): 211—217 2013
- [5] Anand.H.Kulkarni, Ashwin Patil R. K. Applying image processing technique to detect plant diseases. International Journal of Modern Engineering Research. Vol.2, Issue.5, Sep-Oct. 2012 pp-3661-3664
- [6] Sabah Bashir, Navdeep Sharma. Remote Area Plant Disease Detection Using Image Processing. IOSR Journal of Electronics and Communication Engineering. ISSN : 2278-2834 Volume 2, Issue 6 2012, PP 31-34
- [7] SmitaNaikwadi, NiketAmoda. ADVANCES IN IMAGE PROCESSING FOR DETECTION OF PLANT DISEASES. International Journal of Application or Innovation in Engineering & Management. Volume 2, Issue 11, November 2013
- [8] Sanjay B. Patil et al. LEAF DISEASE SEVERITY MEASUREMENT USING IMAGE PROCESSING. International Journal of Engineering and Technology. Vol.3 (5), 2011, 297-301

- [9] Piyush Chaudhary et al. Color Transform Based Approach for Disease Spot Detection on Plant Leaf. International Journal of Computer Science and Telecommunications. ,Volume 3, Issue 6, June 2012
- [10] Arti N. Rathod, Bhavesh Tanawal, Vatsal Shah. Image Processing Techniques for Detection of Leaf Disease. International Journal of Advanced Research in Computer Science and Software Engineering. Volume 3, Issue 11, November 2013
- [11] S.Beucher, F.Meyer. The morphological approach to segmentation: The watershed transforms. Mathematical Morphology Image Processing, E. R. Dougherty, Ed. New York Marcel Dekker, 1993, vol. 12, pp. 433–481.
- [12] B. Bhanu, S. Lee, J. Ming. Adaptive image segmentation using a genetic algorithm. In IEEE Transactions on Systems, Man and Cybernetics, volume 25, pages 1543–1567, Dec 1995.
- [13] B. Bhanu, J. Peng. Adaptive integrated image segmentation and object recognition. In IEEE Transactions on Systems, Man and Cybernetics, Part C. volume 30, pages 427–441, November 2000.
- [14] Keri Woods. Genetic Algorithms: Colour Image Segmentation Literature Review. July 24, 2007.
- [15] Venkatesh Vijayaraghavan, Akhil Garg, Chee How Wong, Kang Tai1 and Yogesh Bhalerao. Predicting the mechanical characteristics of hydrogen functionalized graphene sheets using artificial neural network approach. Journal Of Nanostructure in Chemistry 2013, 3:83
- [16] Akhil Garg, Ankit Garg and K. Tai .A multi-gene genetic programming model for estimating stress-dependent soil water retention curves. Computational Geosciences,2014:p.1-12.
- [17] Akhil Garg, AnkitGarg ,K.Tai and S.Sreedeep. An integrated SRM-multi-gene genetic programming approach for prediction of factor of safety of 3-D soil nailed slopes. Engineering Applications of Artificial Intelligence, 30(2014)30–40.
- [18] Fred Glover .Tabu search for nonlinear and parametric optimization (with links to genetic algorithms). Discrete Applied Mathematics 49 (1994) 231-255.
- [19] Al-Bashish, D., M. Braik and S. Bani-Ahmad, “Detection and classification of leaf diseases using Kmeans-based segmentation and neural-networks-based classification”, Inform. Technol. J., 2011, 10: 267-275.
- [20] Turlapati, Venkata Pavan Kumar, and Manas Ranjan Prusty. ”Outlier- SMOTE: A refined oversampling technique for improved detection of COVID-19.” Intelligence-based medicine 3 (2020): 100023, doi: 10.1016/j.ibmed.2020.100023.