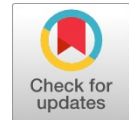# Disease Identification in Chilli Leaves using Machine Learning Techniques

Sufola Das Chagas Silva Araujo, V S Malemath, K. Meenakshi Sundaram

Abstract: Crop diseases reduce the yield of the crop or may even kill it. Over the past two years, as per the I.C.A.R, the production of chilies in the state of Goa has reduced drastically due to the presence of virus. Most of the plants flower very less or stop flowering completely. In rare cases when a plant manages to flower, the yield is substantially low. Proposed model detects the presence of disease in crops by examining the symptoms. The model uses an object detection algorithm and supervised image recognition and feature extraction using convolutional neural network to classify crops as infected or healthy. Google machine learning libraries, TensorFlow and Keras are used to build neural network models. An Android application is developed around the model for the ease of using the disease detection system.

Keywords: Component, formatting, Plant, symptoms.

I. INTRODUCTION

In this paper we describe an application of supervised image recognition and feature extraction using convolutional neural network and object detection algorithm Yolo9000 [1]. The application is Plant Disease Detection. We know that agriculture is backbone of Indian economy, it needs to grow and evolve in its practices. Technology has played important role in evolving agriculture practices since its beginning. Sometimes complex technologies and knowledge are unavailable to farmers.

Disease detection in crops requires various chemical tests, careful observations and laboratory equipment. With our approach farmer can just take picture of a crop and the model will predict if the crop has disease or not. Extensive work has been done in this field. We have surveyed some of the methods to detect plant diseases. Next section describes some of these methods.

II. RELATED WORK

In [2] it is said that most of the symptoms are seen on the leaves, stems and the fruit. Image acquisition, preprocessing, segmentation, feature extraction in image and detection and classification using artificial neural network.

Sufola Das Chagas Silva Araujo, Head of department in Padre Conceicao College of Engineering, Goa, India.

Dr. V S Malemath, Professor, Department of Computer Science and Engineering, KLE Dr. M. S. Sheshgiri College of Engineering and Technology, Belagavi, Karnataka, India.

Dr. K. Meenakshi Sundaram, Faculty, Department of Information and Communication Engineering, Graduate Studies and Research, Botswana.

Images are captured using camera in RGB space. Preprocessing involves manually clipping and cropping image to get interested image region. Whereas in our method object detection is done by neural network. In this method the enhancement of the image is done by increasing the contrast.

Then histogram equalization is done which distributes the intensities of the image to enhance images. Features are then extracted using color, texture or morphology and classified using an ANN or BPNN algorithm in a recurrent neural network.

In the paper [3] authors proposed their system which takes a cropped image of the infected leaf from the user and processes the image. Their system is focused on 5 diseases, Alternaria alternate, Anthracnose, Cercospora, bacterial blight and leaf spot. Also, this system provides voice navigation. It consists of image preprocessing where the image is enhanced by removing noise. After this the image is transformed from the RGB color model to Lab color model to help cluster the segmented images easily. Next the image segmentation where the images are segmented into clusters using k-means and Otsu classifier. Further they perform feature extraction based on the shape and texture. Then finally the disease is identified. The project provides a solution to overcome from the leaf diseases and it also analyze the overall percentage of the affected leaf and its surrounding region.

Reference [4] proposed a system which is capable of detecting the disease from an image of plant leaf and even a fruit using MATLAB image processing. The structure of the system consists of image acquisition, image pre-processing, image segmentation, feature extraction, training classification and recognition. In image pre-processing they used various MATLAB code to resize image, to enhance contrast and RGB to grayscale conversion for further operations like creating clusters in segmentation. In segmentation they converted the image from RGB to Lab format and used K-means clustering method for partitioning of images into clusters in which at least one part of cluster contain image with major area of diseased part. The input image is partitioned into three clusters for good segmentation result. Feature extraction is carried out using Gray level co-occurrence matrix (GLCM). Training and classification are done using Support vector machine (SVM) and also K-means clustering. The disadvantage of SVM is that it supports only binary classification. This can be overcome by further manipulation which the authors of this project have made in order to carryout classification of multiple classes. Finally, the data is stored in the database.

All the methods used in various papers perform preprocessing which is vital to the problem and we shall use similar preprocessing - histogram equalization and brightness variations. The images are cropped manually in most cases which limits the number of leaves that can be examined at a time and requires manual work as well. Our system uses object detection algorithms to automatically detect the leaves.

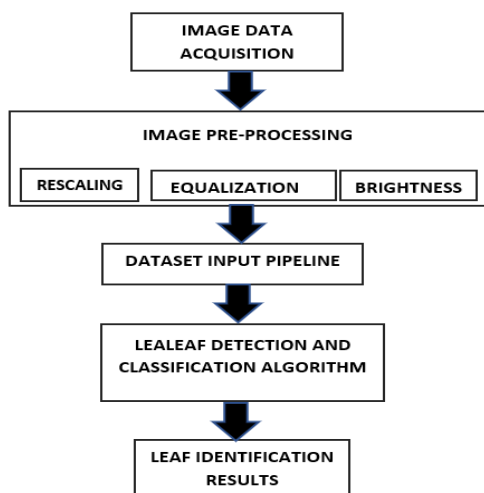### III. PLANT DISEASE DETECTION MODEL

Fig. 1. System Overview

The above schematic diagram is the overall overview of out leaf disease detection model. The steps involved are:

1. Data acquisition: Building raw dataset using videos and still images. This step also involves annotating the images for the purpose for supervised object detection.

2. Image Processing: This step involves performing histogram equalization, rescaling images, normalizing images, changing its brightness and performing rotations and flipping.

3. Data input pipeline deals with building true labels for images based on the configuration of the model. Data pipeline is also used to build batches of images in order to facilitate batch training.

4. Leaf detection and classification algorithm involves convolutional neural network and custom loss function to train the network. The detection algorithm is based on [1].
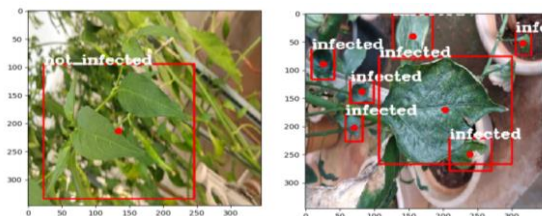
A. *Data Acquisition*

Fig. 2. Labelled Samples from Dataset

As a part of dataset acquisition, we built dataset of infected and healthy G4 species of chilies. To build large dataset we have captured the video of crops and sampled video frames to obtain still images. Thus, for a same sample of plant we have slight variation with respect to the angle of shot.

Next, this dataset was labelled using labelimg tool. By labelling means drawing the bounding boxes around the leaves and classifying them as infected or not_infected. Bounding boxes are stored in the format [xmin, ymin] and [xmax, ymax], which are top-left and bottom-right points of a bounding box. Multiple leaves have been labelled in a single image. Annotations are stored in PASCAL VOC format XML file.

The dataset was split into train and test data. The summary of data is given in Fig. 3

Fig. 3. Number of bounding boxes in dataset

B. *Image Processing*

Image processing involves increasing/decreasing the brightness of the image, performing histogram equalization so as to uniformly distribute the intensities. This is achieved using OpenCV library in python. To perform histogram equalization first the image in converted into Lab space from RGB. Then histogram equalization is performed and image is again transformed into RGB space from Lab space.

We randomly change the brightness of an image within the certain range to achieve data augmentation on the fly. Data augmentation is the method of virtually increasing the dataset size. We also perform random flipping of training images and bounding boxes about x axis, y axis and x-y axis. Random rotations with multiples of 90 are also performed to achieve data augmentation. The image's tensor values are normalized to interval [0, 1] by performing scalar division by 255. The image is also scaled to the input size of the model. Although we have shown image processing as separate strep, we perform image processing on-the-fly in our data input pipeline. This allows us control the options of performing histogram equalization and the range by which we want to vary the brightness.

C. *Estimating Anchor Box Dimensions*

Our model is based on object detection model mentioned in [1] and since [1] uses anchor boxes to generate bounding boxes we need to find best estimates of anchor box dimensions that fits our dataset using K-means clustering. We implemented Mini Batch K-means using following distance function as given in [1].

Distance (box, centroid) =1-IOU (box, centroid)

D. Data Input Pipeline

Data input pipeline loads an image from the disk and its corresponding annotation XML file.

The output of this pipeline is the image tensor and a true tensor of shape

$$GRID_{size}, GRID_{size}, N_{anchors} \times (5 + N_{classes})$$

We find the grid cells in which center of ground truth boxes lie. Such grid cells are said to be containing an object. For each such cell we generate $N_{anchors}$ anchor boxes of defined width and height (estimated using K-means). We compute IOU of ground truth box lying in that grid cell with every k^{th} anchor box generated in that grid cell. This IOU is the confidence score c of that bounding box containing an object with respect to k^{th} anchor box responsible for its prediction. We also encode 4 coordinates ($x_{center}, y_{center}, width, height$) of the ground truth box. Each ground truth box has class label which is represented by one-hot vector encoding. If there are $N_{classes}$ then size of one-hot vector is $N_{classes}$ for each bounding box. Thus, for each ground truth box g containing an object, in cell (i, j) and k^{th} anchor box being responsible for its prediction we have vector of size $5 + N_{classes}$ represented as

$$[x_{center}, y_{center}, width, height, c_{gk}, C_0, C_1, \dots, C_{N_{classes}-1}]$$

$N_{anchors}$ Such vectors are generated for each grid cell containing an object. Note that

$$c_{gk} = IOU(g^{th} \text{ ground truth box}, k^{th} \text{anchor box})$$

is confidence score of containing an object. $C_0, \dots, C_{N_{classes}-1}$ in a vector representing a class.

Out of $N_{anchors}$ boxes in a grid cell, 1 box having highest c_{gk} is considered responsible for detecting object in that grid cell. And this box is used to compute loss during training.

E. CNN Model

The base network model is VGG-16 [5] from which densely connected layers and last max pooling layer were removed. Six convolutional layers randomly initialized were added. A custom output layer which performs direct location prediction as mentioned in [1] was added. This custom layer also computes SoftMax probabilities over classes. The convolutional layers added were of following specifications

Table- I: Layers added to the base VGG-16

Filter size	Strides	Output channels	Padding	Activation
3×3	$(1,1)$	256	VALID	ReLU
3×3	$(1,1)$	256	VALID	ReLU
3×3	$(1,1)$	256	VALID	ReLU
3×3	$(1,1)$	256	VALID	ReLU
3×3	$(1,1)$	128	VALID	ReLU
1×1	$(1,1)$	$N_{anchors} \times (5 + N_{classes})$	VALID	Linear

Output layer of the network performs following direct location predictions as given in [1] and SoftMax predictions over N classes. We trained our model using Adam optimizer.

The loss function used to train our model is similar to loss function defined in [6]. But here we consider class probabilities of all the $N_{anchors}$ boxes predicted per grid cell to compute the loss. We consider only those boxes from predictions which correspond to the boxes that are responsible for predicting bounding box, in ground truth tensor for computing the loss.

Thus, error function we used is as follows,

$$loss = \lambda_{coord} \times loss_{regression} + loss_{classification}$$

$$loss_{regression}$$
$$= \sum_{i=0}^{GRID_{SIZE}^2} \sum_{j=0}^{N_{anchors}} 1_{ij}^{obj} [(x_i - x_i^*)^2$$
$$+ (y_i - y_i^*)^2] + \sum_{i=0}^{GRID_{SIZE}^2} \sum_{j=0}^{N_{anchors}} 1_{ij}^{obj} \left[\left(\sqrt{w_i} - \sqrt{w_i^*} \right)^2 + \left(\sqrt{h_i} - \sqrt{h_i^*} \right)^2 \right]$$

$$loss_{classification}$$
$$= \sum_{i=0}^{GRID_{SIZE}^2} \sum_{j=0}^{N_{anchors}} 1_{ij}^{obj} [c_i - c_i^*]^2 + \lambda_{noobj} \sum_{i=0}^{GRID_{SIZE}^2} \sum_{j=0}^{N_{anchors}} 1_{ij}^{noobj} [c_i - c_i^*]^2$$
$$+ \sum_{i=0}^{GRID_{SIZE}^2} \sum_{j=0}^{N_{anchors}} 1_{ij}^{obj} \sum_{c \in classes} \left(p_{ij}(c) - p_{ij}^*(c) \right)^2$$

Table- II: Model hyper parameters

Model Parameter	Description	Value
$image_w$	Image input width	346px
$image_h$	Image input height	346px
$GRID_{size}$	Grid size	13×13
$BATCH$	Input batch size	3
$ANCHORS$	Anchor box dimensions	(0.3669517,0.45720006) (0.17418965,0.24217035) (0.08529647,0.12489262)
$N_{anchors}$	Number of anchor boxes used	1
$CLASSES$	Classes over which classification is done	{infected, not_infected}
$N_{classes}$	Number of classes	2
λ_{coord}	Scaling for bounding box loss	1.0
λ_{noobj}	Scaling for no object loss	1.0
$EPOCHS$	Number of epochs used to train the network	16

IV. RESULTS

A. Anchor Box Estimates

We used bounding boxes only from the training samples to get estimate of anchor box dimensions. We ran K-means for different values of K (number of clusters/anchor boxes). We summarized the results of K-means below.

Table- III: K V/S Average IOU

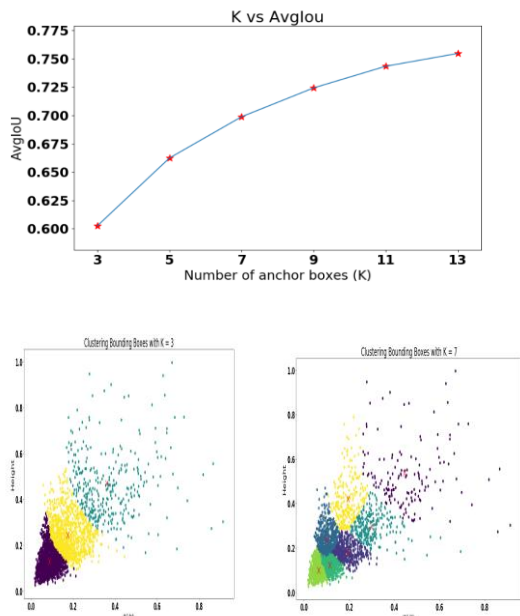| K (number of clusters/anchor boxes)	Avg IOU
3	0.6
5	0.66
7	0.7

Fig. 4. Results of K-means to compute anchor boxes

We see that average IOU over entire dataset increases as we increase number of anchor boxes (as we increase K) which is expected behavior as mentioned in [1]. If greater number of anchor boxes are used in the model, greater will be the accuracy predictions of bounding box coordinates. However, training time is also higher as we increase number of anchor boxes.

B. *Model Evaluation*

The bounding boxes generated by YoloV2 were pruned by using non-maximal suppression algorithm, in which bounding boxes below certain confidence threshold are discarded and bounding boxes overlapping each other, determined by IOU metric, are discarded. The detections on few samples are shown in Fig 5.
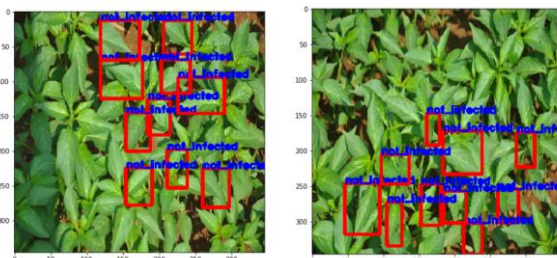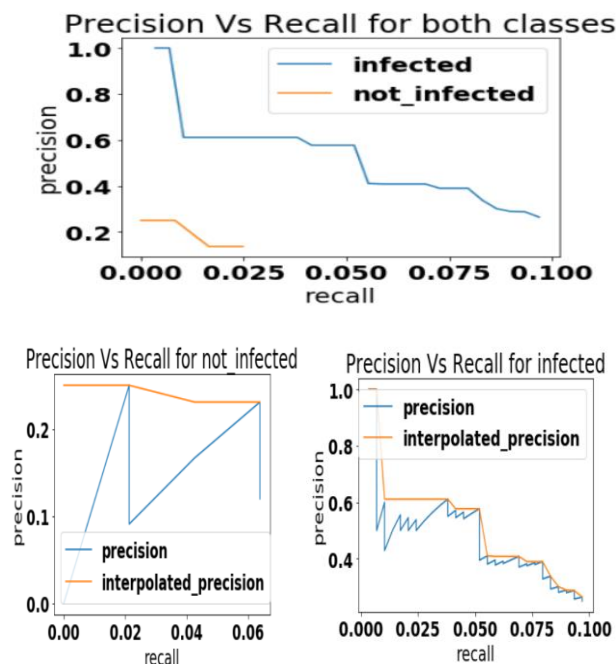

Fig. 5. Detections using our model

Fig. 6. Detections using our model

Table- IV: mAP of plant disease detection model at IOU=0.5

Dataset	mAP for infected	mAP for not infected	Overall mAP
Test	65.27%	17.60%	41.43%
Train	46.28%	76.71%	61.49%

The model was evaluated using metric mean average precision (mAP) for infected and not infected class. We evaluated mAP for IOU = 0.5. The mAP for train dataset is 61.49%. Following table summarizes mAP for train, test data and whole data

REFERENCES

1. J. R. a. A. Farhadi, "YOLO9000: Better, Faster, Stronger," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017.
2. S. D. Khirade and A. B. Patil, "Plant Disease Detection Using Image Processing," in International Conference on Computing Communication Control and Automation, Pune, 2015
3. S. G, Dhivya., R, Latha.S and R.Rajesh, "PLANT DISEASE DETECTION AND ITS SOLUTION USING IMAGE CLASSIFICATION," International Journal of Pure and Applied Mathematics, vol. 119, no. 14, pp. 879-883, 2018.

328

4.   K. Singh, M. &. Chetia, S. &. Singh and Malti, "Detection and Classification of Plant Leaf Diseases in Image Processing using MATLAB," International Journal of Life sciences Research, vol. 5, no. 1, pp. 120-124, 2017.

5.   K. Simonyan and A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, arXiv: 1409.1556v6 [cs.CV], 2014.

6.   J. Redmon, S. Divvala and R. G. a. A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," J. Redmon, S. Divvala and R. G. a. A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016.

7.   Y. Lecun, L. Bottou and Y. B. a. P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov 1998.

8.   S. Ren, K. He and R. G. a. J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 2017.

9.   R. Girshick, "Fast R-CNN," in 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, 2015.

10. Sufola Das Chagas Silva Araujo, Meenakshi Sundaram Karuppaswamy, "Comparative Analysis of K-Means and K-Nearest Neighbor Image Segmentation Techniques" in IEEE 6th International Conference on Advanced Computing (IACC)Year:February2016,DOI: 10.1109/IACC.2016.27,IEEE Conference Publications.

11. Sufola Das Chagas Silva Araujo, Meenakshi Sundaram Karuppaswamy "Vegetable-Fruit Identification Based on Intensity and Texture Segmentation" in special issue of SCOPUS INDEX JOURNAL titled "International Journal of Control Theory and Applications (2016)".(ISSN : 0974-5572), 12th October, 2016.

12. Sufola Das Chagas Silva Araujo, Meenakshi Sundaram Karuppaswamy "An optimized approach for enhancement of

13. medical images" in Biomedical Research 7th September 2016, "An International Journal of Medical Sciences", THOMPSON REUTERS INDEX JOURNAL,(ISSN 0970-938X),Special Issue: S283-S286'

14. Sufola Das Chagas Silva Araujo, Meenakshi Sundaram Karuppaswamy "Chest CT Scans Screening of COPD based Fuzzy Rule Classifier Approach " in International Conference on "Advance in Signal Processing and Communication-SPC 2016",July 9-10,2016,Proceedings published by WALTER De GRUYTER

15. Sufola Das Chagas Silva Araujo, Meenakshi Sundaram Karuppaswamy "Recognition and Detection of Object Using Graph-Cut Segmentation " in " 7th World Conference on Applied Sciences, Engineering & Management (WCSEM)" organized by American Business School of Paris, France during 26-27 October 2018. Proceedings ISBN 13: 978-81-930222-3-8 ,SCOPUS Indexed International Journal of Engineering and Technology, 2227-524X

16. Sufola Das Chagas Silva Araujo, Meenakshi Sundaram Karuppaswamy,V. S. Malemath "Detection and Segmentation of Blood Cells Based on Supervised Learning" in"IndianJournalofScienceandTechnology,Vol12(20),DOI:10.17485/ijst/2019/v12i20/145114, May 2019

AUTHORS PROFILE

Sufola Das Chagas Silva Araujo is perusing her doctoral degree in Department of Computer Science Engineering, Visvesvaraya Technology University, India. She obtained her Master's degree in Information Technology, Computer department from Goa University, India. She got her Bachelor's degree in Electronics and Telecommunication Engineering from Goa University, Goa. She is currently working as a faculty and Head of department in Padre Conceicao College of Engineering, Goa, India. Her specializations include Machine Learning, Digital Image Processing, Image Processing, Pattern Analysis, Computer vision and Computer Networks. Her area of research are Pattern Analysis and Machine Learning. she has put forth 17 + years of Teaching and Industrial Experience. She has more than 15 publications in reputed International Journals and Conferences. She is life member of professional body like ISTE.

Dr. V S Malemath, is a Professor in the Department of Computer Science and Engineering, KLE Dr. M. S. Sheshgiri College of Engineering and Technology, Belagavi, Karnataka, India. He earned a Bachelor Degree in Electronics and Communication Engineering from KLE College of Engineering and Technology, Belagavi, and a Master Degree from BITS Pilani, India. PhD in Computer science from the Department of Studies in Computer science, Gulbarga University, Karnataka, India.His research areas are Image Processing and Pattern Recognition, Character Recognition, soft computing, Internet of things and Graph Theoretic techniques. He has more than 90 publications in reputed International conferences and Journals. He is life member of professional bodies like ISTE and IETE.

Dr. K. Meenakshi Sundaram, Dr.K.Meenakshi Sundaram obtained his Doctoral Degree at Faculty of Information and Communication Engineering, Anna University, Chennai. He obtained his Master's degree in Computer and Information Technology from Manonmaniam Sundaranar University, Tirunelveli. He got his Bachelor's degree in Electrical and Electronics Engineering from Madurai Kamaraj University, Madurai. Currently, he is working as a Fellow in Faculty of Graduate Studies and Research, Botswana. His specializations include Digital Image Processing, Software Engineering, Software Testing and Computer Networks, His current area of research interests are Medical Imaging Techniques and Information Security. He has put forth 19 + years of Teaching and Industrial Experience. To his credit he has published papers in 15 International Journals and 7 International Conferences.