Jurnal Informatika
Vol. 17., No. 1, January 2023, pp. 39-56

ISSN 1978-0524 (print) | 2528-6374 (online)
http://journal.uad.ac.id/index.php/JIFO/index

A Hybrid Approch Tomato Diseases Detection At Early Stage

Arif Ullah [a,1,*], Muhammad Azeem khalid [b], Dorsaf Sebai [c], Tanweer Alam [d], Canan Batur Şahin [e]

[a] School of Computing,Riphah International University, Faisalabad
[b] Faculty of Computing and Informatics Multimedia University Cyberjaya, Selangor Malaysia
[c] National Institute of Applied Sciences and Technology Tunis, Tunisia
[d] Islamic University of Medina,Department of Computer Science
[e] Turgut Özal University department of Software Engineering
[1] arifullahms88@gmail.com
* Corresponding Author

Received 26 October 2022; accepted 28 December 2022; published 13 January 2023

ABSTRACT

In traditional farming practice, skilled people are hired to manually examine the land and detect the presence of diseases through visual inspection, but the visual inspection method is ineffective. High accuracy of disease detection is one of the most important factors in crop production and reducing crop losses. Meanwhile, the evolution of deep convolutional neural networks for image classification has rapidly improved the accuracy of object detection, classification and system recognition. Previous tomato detection methods based on faster region convolutional neural network (RCNN) are less efficient in terms of accuracy. Researchers have used many methods to detect tomato leaf diseases, but their accuracy is not optimal. This study presents a Faster RCNN-based deep learning model for the detection of three tomato leaf diseases (late blight, mosaic virus, and leaf septoria). The methodology presented in this paper consists of four main steps. The first step is pre-processing. At the second stage, segmentation was done using fuzzy C Means. In the third step, feature extraction was performed with ResNet 50. In the fourth step, classification was performed with Faster RCNN to detect tomato leaf diseases. Two evaluation parameters precision and accuracy are used to compare the proposed model with other existing approaches. The proposed model has the highest accuracy of 98.6% in detecting tomato leaf diseases. In addition, the work can be extended to train the model for other types of tomato diseases, such as leaf mold, spider mites, as well as to detect diseases of other crops, such as potatoes, peanuts, etc.

KEYWORDS
Faster
Region-Based
Convolutional
Neural Network
Fuzzy
C Means

1. Introduction

Diseases are the most infected area of crops and plants, which can significantly reduce the quality of products. The most important thing is to preserve the plants so that they are necessarily checked for the presence of diseases at the early stages. Day by day, the increase the plant diseases are very harmful to humans, and the prices of market food are also rising. In addition, it can affect the quality of food. According to global diseases affecting factories, the costs are 20-40 percent, which is estimated to cost $220 billion in product quality loss [1].To meet the food needs of the world's growing population, it is necessary to increase food production and minimize food losses. It is estimated that approximately 9.1 billion people will live in the world by 2050, and meeting their food needs will not be an easy task. Food production should increase by more than 70% from the current level. To achieve this goal, the agricultural industry is increasingly using chemicals, such as disinfectants and fungicides, to control pests and plant diseases. To control and manage the spread of diseases of agricultural crops, there is a need to develop new safe methods for early diagnosis and detection of diseases [2]. Pakistan is an agricultural country, and the land is cultivated using old production technologies, which leads to various structural problems. In agriculture,

Jurnal Informatika
Vol. 17., No. 1, January 2023, pp. 39-56

disease recognition is a serious problem. This study focuses on the identification and classification of diseases of tomato leaves. Therefore, it is necessary to take appropriate measures to combat diseases of agricultural products while reducing the use of chemicals for disease control and losing money [3].Most diseases are caused by viruses, bacteria and fungi contained in the plant. The identification and detection of these plant leaf diseases are important factors in preventing the spread of diseases and managing them. The various methods used to identify and detect these plant diseases are based on the knowledge of farmers. Now day's deep learning has become the most important method used to diagnose various features of crops in the field of agriculture, as it minimizes confusion and helps the expert avoid abuse in the diagnosis and control of plant diseases that in the old days affected society and the world. Plants are the most important assets of an Asian country, so the loss of plant quality has affected the overall economy of the country [4], [5].Agriculture has a significant impact on the country's economy, and is also the basis of the lives of individuals. Recently, plant diseases such as microbes, late blight, leaf spotting, tomato mosaic and yellow rot have also been widespread, which seriously affect the development of plants and lead to a decrease in the value and quantity of products [6].Plants are affected by a wide variety of pests and diseases, especially in very hot and subtropical regions of the world. Plant disease involves a complex interaction between the host plant, its carrier and the disease. The formulation of this problem is sometimes identified with the consequences of environmental changes in the air and how this changes the biological system [7]. Environmental change significantly affects territorial environmental factors, such as temperature, diseases and precipitation, which subsequently serve as a carrier in which microbes, diseases, can destroy crops, and thus have a direct impact on the population, for example, on money, well-being and impact on work [8].

2. Related Work

Supervised learning is a method of developing artificial intelligence (AI) that involves training a computer algorithm on labelled input data for a specific output. When presented with previously unseen data, the model is trained until it recognises the underlying patterns and relationships between the input data and the output labels, allowing it to produce accurate labelling results. Tomatoes are a vital crop that is grown all over the world. It is susceptible to a variety of diseases, reducing tomato quality and yield while also causing significant economic losses [9]. Tomato grey leaf spot is a common disease that destroys and damages tomato leaves, preventing them from growing and producing fruit. The Fuzzy Support Vector Machine (Fuzzy-SVM), Convolution Neural Network (CNN), and Region-based Convolution Neural Network (RCNN) are used in this study to identify tomato plant leaf disease (R-CNN). Images of tomato leaves with six diseases and healthy samples were used to confirm the findings. Tomato leaf disease datasets are classified into two categories: training data (70 percent) and testing data (30 percent) (30 percent). Tomato Plant Disease is a dataset. The Bacterial Spot, Mosaic Virus, Yellow Leaf Cur Virus, Early Blight, Late Blight, Leaf Mold, and Healthy classes are included in the Tomato Plant Disease dataset. R-CNN classifiers are used to categorize the type of illness. The confusion matrix is a metric used to assess the performance of a machine learning classification task with two or more output classes. This table contains four distinct combinations of projected and actual values. It's excellent for evaluating recall, precision, specificity, accuracy, and, most importantly, AUC-ROC curves. Fuzzy SVM and CNN classification methods are analyzed and compared with R-CNN to determine the most accurate model for plant disease prediction. When compared to other classification methods, the R-CNN-based Classifier has the highest accuracy of 96.735 percent.

The major factors that lessen food production include weeds, climatic changes, plant diseases, and so on. Extensive data shows that 80% of the food production is produced by small-scale farmers in developing countries like Pakistan and similarly expressed that 50% of the yield reduction occurred by the severity of pests and diseases [10]. This work aims to develop an automated mechanism for detecting diseases on cultivated land using advanced image processing techniques and algorithms. Deep learning techniques are used to detect and classify tomato disease in plants, specifically with the deep detector: Faster R-CNN with deep feature extractor: ResNet50. We trained and tested the proposed system using our tomato dataset, which contains 1090 images of early, medium, and late stages of tomato disease. Even in complex plant surrounding areas, our proposed system detects early blight, leaf curl, septoria leaf spot, and bacterial spot of tomato disease. The proposed system detects four types of tomato diseases in tomato crops: early blight, leaf curl, septoria leaf spot, and bacterial spot. Additionally, work can be extended.by training the model with other types of tomato diseases i.e late blight, leaf mold, spider mites, and so on, and also to detect diseases for other crops like potatoes, peanuts, and so on. The Faster R-CNN-based Classifier has the most remarkable accuracy of 80.0952 percent compared to the other classification approaches [11].

Tomatoes can get many diseases at every stage of the cultivation process depending on environmental and climatic factors [12].Tomato growers sometimes struggle to monitor leaves to detect these diseases. As a result, deep learning-based systems for detecting diseases have been developed. It has demonstrated its worth by solving problems in a variety of fields, including classification. A convolutional neural network (CNN) is particularly effective in image classification, recognition, and detection. In this article, a hybrid-based CNN model for the classification of diseases on tomato leaf images is proposed. The performance of the method has been examined by the tomato leaf disease detection dataset and Taiwan datasets. For the extraction of features, well-known CNN architectures such as AlexNet, ResNet50, and VGG16 are used at first. The feature transfer method extracts features from the last fully connected layers of the architectures. After that, the minimum redundancy maximum relevance feature selection algorithm is applied to these features for optimization. The features gathered are concatenated. Concatenating features are classified by popular machine learning classification algorithms. With the proposed method, the highest performance values for the tomato leaf disease detection and Taiwan dataset show an accuracy of 98.3% and 96.3%, respectively [13].

In this study, a lightweight CNN with 20 layers and reduced trainable parameters was designed using the ResNet topology [14]. Then, the commonly used attention modules, namely the convolutional block attention module (CBAM), self-attention module, squeeze-and-excitation module, and the dual attention module, were integrated into the base network to observe the impact of different attention mechanisms on conventional CNN. Moreover, the performance of the models with and without attention mechanisms was assessed by employing well-known classification metrics (accuracy, precision, recall, and F1 score). All the models were trained, validated, and tested using tomato disease datasets split at a ratio of 8:1:1 for training, validation, and testing. Furthermore, the productive number of attention modules and their locations in the base network were comprehensively assessed through an ablation study. Finally, the computational complexity of the models, the training and testing time per image, network parameters, and sizes were calculated and compared parametrically [15]. Therefore, the main objectives of this study were to design a lightweight and computationally efficient network for the classification of a few classes of plant diseases, improve the performance of conventional CNN by amending it with an attention mechanism, and identify an effective and efficient attention module for plant disease detection. Ten classes of tomato leaf images (9 diseases and one healthy) that were part of the Plant Village public datasets were collected. This study experimented with various attention modules and analyzed their performance in tomato disease classification. Attention modules used for different purposes were employed. The network architecture, computational complexity, and performance were comprehensively compared [16].

3. Proposed Method

The Faster RCNN have one of those approaches which are integral to understand the to the proposals of districts since the time these idea regions are showcased into the diseases categories arranged by CNNs and begun to ending in the technique of Faster RCNN. The accuracy of the results is based on demonstrating the idea of space module. This method is not fixed sized to look at the infected images of the sugarcane. The length and the width should be must restricted in the image input along with the lines forestalling and twisting. The speed was also upgraded along with the upgraded Regional Proposal Network shortly known as RPN. In most of the cases this method is led by the Faster RCNN for the purpose

Jurnal Informatika
Vol. 17., No. 1, January 2023, pp. 39-56

of assembling a solitary and bounded model which is made up of RPN and the Faster RCNN [17].The Faster RCNN and the RPN are seen as a module of provincial age network which utilizes the RPN and it is also a suggested proposal based on the algorithm. A position is relapsed when the fully associated layers which are prepared happen on the map component and the predefined features are applied to identify the objectives. The choice of the area is based on the intensity of the disease and the area is chosen where the disease is on the peak. The accommodative box displays the sources of the Faster RCNN which are used to get the ideal model of the location of the tomato leaf disease. In general, the images of the tomatoes consisted of different levels of surfaces which are utilized to the components of the images and led to the equivalent images of tomato. Both high and low resolutions are used during this process as the higher one is used to take gander for focusing on the little things and the other is used to differentiate the astronomical articles [18]. The procedures are examined the primary explanation which is behind the scenes to multiple objects and caused crumbling wavelet. The higher resolution images are mathematical in particularly and applied for the purpose of restricting collectively and mixing the assorted pictures during the working procedures. The Faster RCNN is used to differentiate the inner objects in the given edges in computing. A CNN spine is used to separate the highlighted maps from the beginning and this method is used to appraise the boxes of bounding. Without considering a specific districts which may contains significant articles it used to execute the ROI pooling layer which is further used to blend the separated components of the maps and also empowers the classifier to yield the article class and the appropriate bounding box [19]. The idea which is behind and essential to the Faster RCNN is to break the signs concerning the study which is done by the scale. A huge load of premium is got while dealing in signals with mathematical method of examination is used. The earlier model of the Faster RCNN is the organization of signs and the examination based on the images. The Fig. 1 in the research framework describes that the first phase is the image input phase. The pre-processing steps consist of background remover, noise, and color space conversions. The raw image is not enough to distinguish between the healthy and infected tomato leaf. After pre-processing the Fuzzy means C Clustering is applied for used segmentation of the images. The Fuzzy C Means is used to enhance the edges of the images. After Segmentation the Resnet50 is applied to use to feature extraction from an input image. The Resnet50 enhances the color and shapes to detect the disease area of the image accurately. For the purpose of the categorization of the healthy and unhealthy images of the collection of the feature extraction passes through the Faster RCNN classifier. For the purpose of measuring the accuracy the complete performance of the method is detected and measured [20].Fig. 1 present Research Framework.

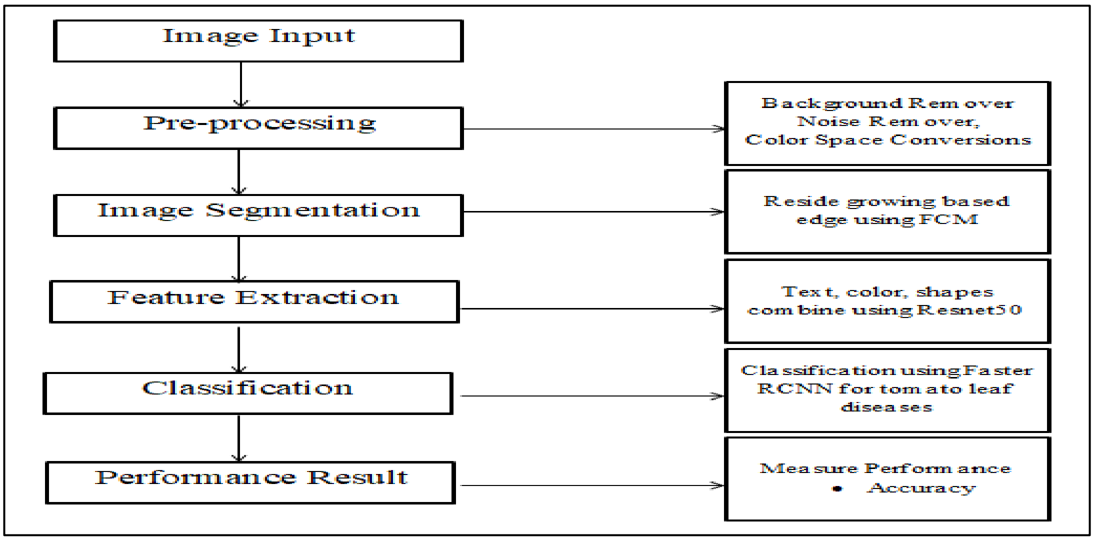

Fig. 1. Research Framework

3.1. Data Collection

The plant village dataset which is consisted of 54,309 marked pictures taken from the 14 distinct yields. For this project we have considered the images of the tomato leaves distributed as follows: 1000 of the diseased leaf with late blight, 1000 of the diseased leaves with the mosaic virus, and 1000 of the diseased leaves with septoria leaf spot, obtaining a total of 3000 images that were used for the classification between healthy and diseased tomato leaf [21]. On the other hand, 3000 images corresponding to diseased leaf were used to classify the types of disease. Test images from the plant village dataset are publicly available on the Kaggle website.

3.2. Data Preprocessing

The pre-processing of the data was used to enhance the quality of the images and to remove all the unwanted background and noise data from the input images. Pre-processing phase is the most important one in terms of to enhance the quality and to remove all irrelevant data from the images so that the only target area of images is covered for segmentation.

3.3. Fuzzy C – Mean Segmentation

Fuzzy C-Means clustering is a specific type of soft clustering method the data is assigned in which an indicating score is pointed to see that whether it belongs to the targeted cluster or not. This process of segmentation is distributed into multiple steps. In the first step the input images are converted into the desired features which is dependent on the on the software which is being used. It is one of the successful methods of data clustering. In clinical diagnosis the successful results are given by the algorithm which is unsupervised. It is further significant in image analysis, recognition of target etc. A segmentation technique which is based on the fuzzy clustering, in enhanced images, is used. The distinguished and very high precision images segmentation is achieved with fuzzy clustering in FCM which is continuously modified. The leaf disease area has been clustered or grouped in the image [22]. The algorithm is composed of the following steps:

- It clusters images in the feature space with addition to the next conditions and the number of clusters is c, while the fuzziness index is m and the stop condition is ε.
- Repeat for each pixel aij of image.
  - It finds out that into which of the cluster CA belong pixel aij at the most.
  - It also finds out that whether in the closest surroundings of pixel aij exists segment Rk , which points belong to same cluster CA.
  - If such segment Rk exists, than pixel aij add to segment Rk , else create new segment Rn and add pixel aij to new segment Rn.
- All the segments are merged which belong to one cluster and are neighbors.
- All segments of the border are also arranged. Fig. 2 present Pseudo Code for FCM.

The pseudo Code for FCM

Manually set the number of clusters C and fuzziness degree M and error ε

Cluster centers $c_i^{(0)}$ are initialized randomly $k = 1$

While $\left\|c_i^k - c_i^{(k-1)}\right\| > \varepsilon$

Using cluster center $c_i^{(k-1)}$ calculate the membership matric $u^k$by:

$$u_{ij}^k \leftarrow \frac{1}{\sum_l^c \left(\frac{d(x_i, c_i^{(k-1)})}{d(x_i, c_l^{(k-1)})}\right)^{2/m-1}}$$

Using membership matric u^k, cluster center c_i^kis update by

$$c_i^k \leftarrow \frac{\sum_{j=1}^N (u_{ij}^k)^m x\ j}{\sum_{j=1}^N (u_{ij}^k)^m} \quad k = k + 1 \text{ [23].}$$

Fig. 2. Pseudo Code for FCM

Jurnal Informatika
Vol. 17., No. 1, January 2023, pp. 39-56

3.4. Feature Extraction

For the proposed of extracting features the ResNet50 is used to get the enhanced quality of the images. It also improves the accuracy of the results while the classification at the complex level is being done. The foundations of the many of the computer visions tasks are served by the Residual Network, shortly known as ResNet. This network enables us to train the deep neural networks having more than 150 layers. Before ResNet the training was very difficult related to the deep network due to the issues of the vanishing gradients. Fig. 3 Flowchart ResNet50 feature extraction and Fig. 4 present Pseudo Code for Resnet50.

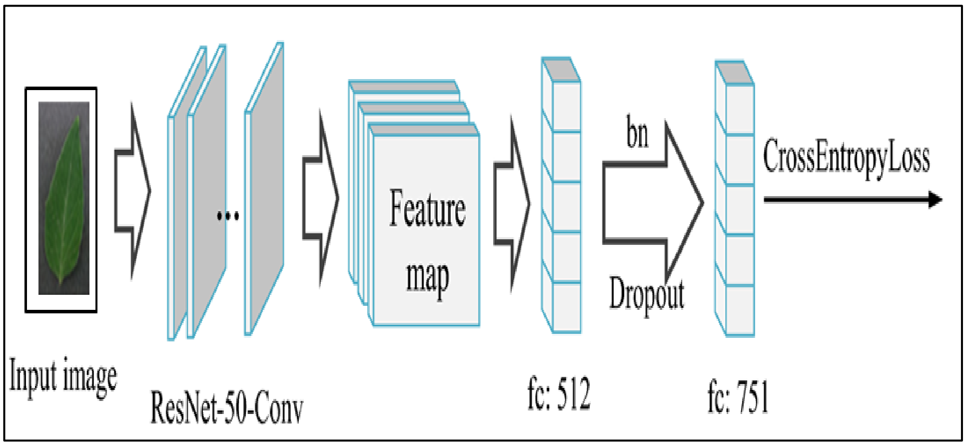

Fig. 3. Flowchart ResNet50 Feature Extraction

```
ResNet50
    # Define the input as a tensor with shape input_shape
    X_input = Input (input_shape)
    # Zero-Padding
    X = ZeroPadding2D ((3, 3))(X_input)
    # Stage 1
    X = Conv2D (64, (7, 7), strides=(2, 2), name='conv1', kernel_initializer=glorot_uniform (seed=0))(X)
    X = BatchNormalization (axis=3, name='bn_conv1')(X)
    X = Activation ('relu') (X)
    X = MaxPooling2D ((3, 3), strides=(2, 2))(X)
    # Stage 2
    X = convolutional_block (X, f=3, filters=[64, 64, 256], stage=2, block='a', s=1)
    X = identity_block (X, 3, [64, 64, 256], stage=2, block='b')
    X = identity_block (X, 3, [64, 64, 256], stage=2, block='c')
    ### START CODE HERE ###
    # Stage 3 (≈4 lines)
    X = convolutional_block (X, f = 3, filters = [128, 128, 512], stage = 3, block = 'a', s = 2)
    X = identity_block (X, 3, [128, 128, 512], stage=3,block='b')
    X = identity_block (X, 3, [128, 128, 512], stage=3,block='c')
    X = identity_block (X, 3, [128, 128, 512], stage=3,block='d')
    # Stage 4 (≈ 6 lines)
    X = convolutional_block (X, f=3, filters = [256, 256, 1024], stage = 4, block='a', s = 2)
    X = identity_block (X, 3, [256, 256, 1024], stage=4, block='b')
    X = identity_block (X, 3, [256, 256, 1024], stage=4, block='c')
    X = identity_block (X, 3, [256, 256, 1024], stage=4, block='d')
    X = identity_block (X, 3, [256, 256, 1024], stage=4, block='e')
    X = identity_block (X, 3, [256, 256, 1024], stage=4, block='f')
    # Stage 5 (≈3 lines)
    X = convolutional_block (X, f = 3, filters = [512, 512, 2048], stage = 5, block='a', s = 2)
    X = identity_block (X, 3, [512, 512, 2048], stage=5, block='b')
    X = identity_block (X, 3, [512, 512, 2048], stage=5, block='c')
    # AVGPOOL (≈1 line). Use "X = AveragePooling2D (...)(X)"
    X = AveragePooling2D ((2,2), name="avg_pool")(X)
    ### END CODE HERE ###
    # output layer
    X = Flatten ()(X)
    X = Dense (classes, activation='softmax', name='fc'+str (classes), kernel_initializer = glorot_uniform (se
    ed=0))(X)
    # Create model
    model = Model (inputs = X_input, outputs = X, name='ResNet50')
    opt = Adam (lr=0.0001)
    model.compile (optimizer = opt, loss='categorical_crossentropy', metrics=['accuracy'])
    return model
```

Fig. 4. Pseudo Code for Resnet50

3.5. Classification

Faster RCNN based on ROI pooling approach. It slices the image features. For the purpose of extracting features in a computationally accurate way it uses a modified form method of spatial pyramid pooling. It also classifies the results of the pictures. It is an advance form of the Fast R-CNN.

- It takes an image and sends it convent which brings back the feature maps for it.
- It would apply RPN on these feature maps to bring the proposals of the object.
- Furthermore, it would be used to reduce the size to bring it at the exact level.
- Lastly, for the purpose of classification and prediction the bounding boxes of the pictures it forwards these proposals to the layer which is fully connected.

Faster RCNN consists of two parts:

- RPN is used for region proposals.
- Fast RCNN is used to detect the object in proposals regions [24].

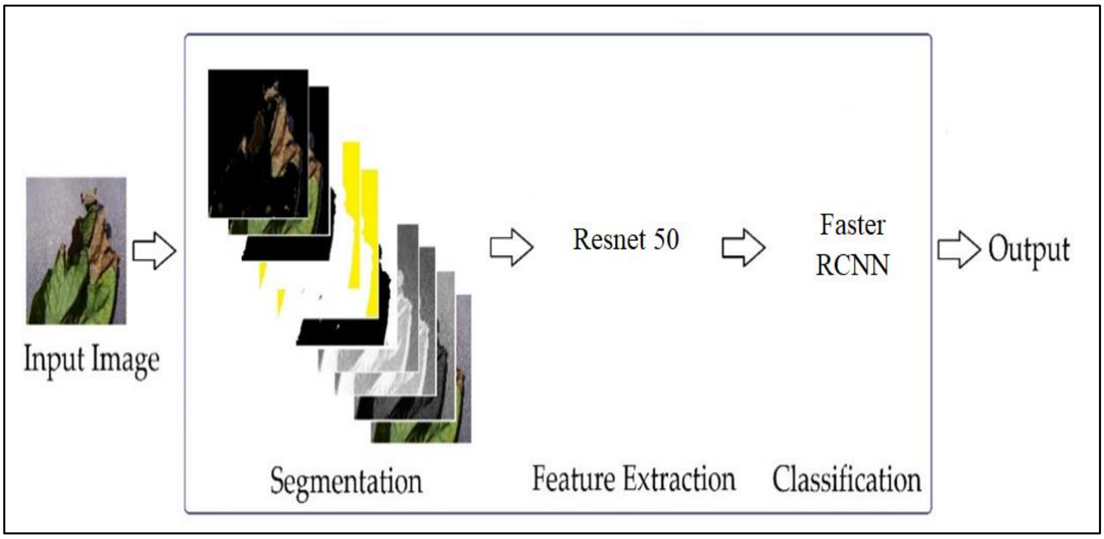

Fig. 5. Flowchart of Faster RCNN Classification

The RPN module is responsible for generating region proposals. It applies the concept of attention in neural networks, so it guides the Fast R-CNN detection module to where to look for objects in the image [25].
The Faster R-CNN works as follows:

- The RPN generates region proposals.
- For all region proposals in the image, a fixed-length feature vector is extracted from each region using the ROI Pooling layer.
- The extracted feature vectors are then classified using the Fast R-CNN.
- The class scores of the detected objects in addition to their bounding-boxes are returned.

```
Initialize the parameters
confThreshold ->0.5              #Confidence threshold for bounding box predictions
maskThreshold -> 0.3              # Mask threshold for binary masks
Load the models
weightsPath -> .../frozen_inference_graph. pb            #pre-trained weights
configPath ->.../rcnn_inception_v2_coco_pets. pbtxt       #text graph file to load model onto OpenCV

Initialize the video stream
vs -> cv2.VideoCapture(.../ weapon_video.mov) #loading the video

Process each frame
grabbed, frame -> vs. read () #reading each frame and returning the coordinates of the frames
blob -> cv2.dnn.blobFromImage(frame) #creation of 4D blob from a frame
net. setInput (blob) #passing the blob as an input to the ConvNets

Extract the bounding box and drawing the box for each detected object
for i in range(numDetections):
        box -> boxes [0, 0, i]
        mask -> masks[i]
            left -> int (frameW * box [3]) #Acquiring bounding boxes
            top ->int (frameH * box [4])
            right -> int (frameW * box [5])
            bottom -> int (frameH * box [6])
cv2.rectangle(frame, (startX, startY), (endX, endY), colour, 2) #drawing bounding boxes
```

Fig. 6. Pseudo Code of Faster RCNN

3.6. Proposed Method

The model based on the Faster RCNN is a methodology which is proposed. It is also displayed in Fig. 5. It stores the information image to the CNN in Faster RCNN which further produce the maps and the convolutional. Guides are used to separate the districts which are recommended. For the purpose of reshaping the proposed districts ROI is used which further fixes the size in order to handle the network in complete connected way. It is the machine learning algorithm. The Regional Proposal Network, also known as RPN, is used by faster RCNN. RPN uses the images in order to highlight the maps of districts as input and outputs slew of recommendations of items, each with having a score at the rate of F1. In a Faster RCNN approach, the following steps are typically repeated:

- In order to obtain the featured map, backbone of the ResNet50 is utilized to pass the images. Apart from that the efficiency of time there is another reason of utilizing RPN as a proposed generator which is used to share the weight of benefits between RPN backbone and the backbone of the Fast R-CNN detector.

- Following that, the RPN bounding box proposals are utilized in order to pool the features from the backbone feature map. The ROI pooling layer is used to handle it. In essence, the ROI pooling layer is being operated by a). The proposal b) is corresponded in order to region the backbone feature map. It divides the region further into sub-windows which are fixed in number and c). The R-CNN learns more related to the ROI layers of pooling and its advances and produces a size which is fixed.

- A size of (N,7,7,512) pooling layer of ROI where the numbers of N having proposal region in terms of algorithm. The larger features are stored into the categories and regressive branches when they are connected.

There are three stages in the generated pipeline in this work to accurately detect the diseases in the different plants like of tomato. Initially, a preprocessing has been done successfully. The ResNet50 is used to feature extraction from object in the image which is given by utilizing the features of the plan of engineering. In order to detect the diseases in the plants the extricated featured maps are used for the purpose of detecting diseases in the tomato plants. According to the Faster RCNN methods in which the CNN spine is generated a target is set in which a poling layer could empower the classifier in order to yield the required output with the help of a regional proposed network with the combination of interest pooling layer. At the end, the different objects related bounding areas coordinated collectively beyond the outline of resulting with the higher chances that the diseases will be detected in the every image taken using the deep sort algorithm [26]. Fig. 7 present the flowchart of Faster RCNN based tomato diseases detection method.

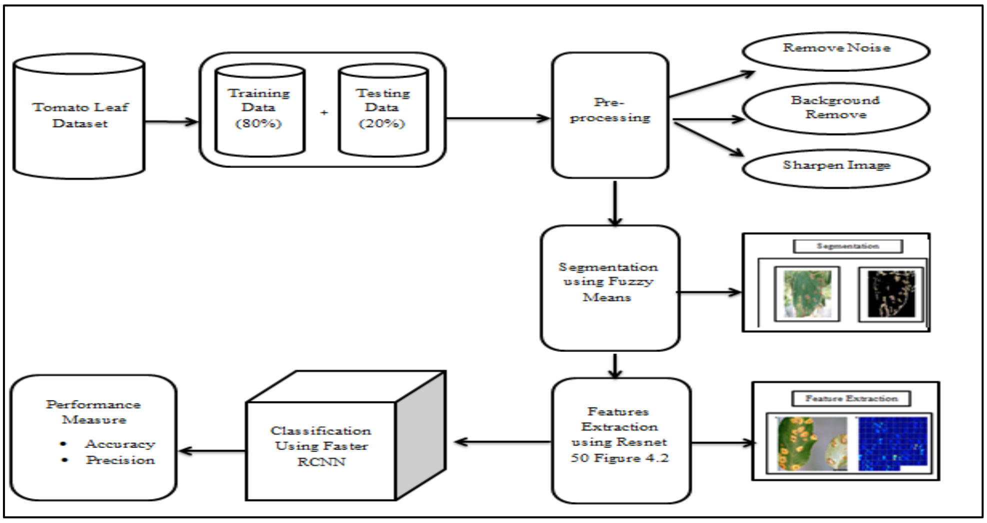

Fig. 7. Flowchart of Proposed Techniques

3.7. Multi-Stream CNN Feature Extractor

A novel multi-stream CNN is at the core of this proposed method where images abased on regular approach of scaling is being used to learn the quality of objects at the various levels and the equal convolutional streams are being used to snub. An item is contained on the info image which is natural. During the convolutional task the various subtleties are abused by a specific part of size in order to investigate the each stream. As a result the every channel is delivered through a specific convolutional which ae caugt by the angles that are missed in the modified stream due to unmistakable portion which is utilized in reporting which is inputting the investigation reasoning of the image. By examining the activity of the convolutional this conduct could be seen. The distinction in the information of the given image I and the internal pixels of the concerned image is separated by I (x,y), where these values (x,y) are compared to the arranged pixel and the shifted pixel which are acquired by the application of the portion, is proceeded through the equation accompanied [27].

$$f(x,y) = w * I(x,y) = \sum_{\delta_x=-i}^{i} \sum_{\delta_y=-j}^{j} w(\delta_x,\delta_y) I(x+\delta_x, y+\delta_y) \tag{1}$$

In this equation w is compared to the loaded piece and the dx and dy are representing the x and y facilitated inside of the bit. After that the i=j=f3, 5,7g are addressing to the segment size. It is evident to get the idea that with the application of the different convolutional activities having multiple sized pieces to the provided image resulted in differently yield channels due to the contrasted size of the portion which is utilized. To get updated with the firsthand proportion view of the outlined which is resized there are three mistake stream of equal size which is which investigates the images using the distinctive part which is estimated as k and possibly delivers highlighted maps having equal shapes. Particularly the stream 1 is contained 10 layers of convolutional with having size of 3 x 3 with the addition of 4 tasks of max-pooling which are loaded after the each couple of convolutions [28]. The stream 2 is consisted of 9 layers with having a partially state of 5 x 5 with the addition of 10th convolution as of k=3 which is used to reach at the exact component of the map size.in order to effectively reduced the size of image there are another 4 further layers of max-pooling after those of the 2nd, 5th, and the 7th. Conclusively, third stream is used to incorporate the 8th convolutions having a size of 7 x 7 like the 10th layers which has a value of k=3 just like the stream 2 for the use of the last information which is decreased. In order to handle this situation four layers of max-pooling are utilized after that of the third, sixth, seventh, and the eighths to reach at the correct size of the element map. In terms of channels number which begins from the size of 64 the authors are multiply after the every possible activity of pooling having

Jurnal Informatika
Vol. 17., No. 1, January 2023, pp. 39-56

computational frequency of 64, 128, and 25 respectively. Everything is taken into consideration after the application of the last pooling in which a bottleneck of the channel is being applied which is carried out the convolutions with 128 channels in order to lessen the boundaries in terms of numbers which are delivered [29].

3.8. Object Detection

Object detection techniques conform to the three major steps indicated in Fig. 8. The first step is to generate a number of region proposals. These region proposals are candidates that may contain objects. The Faster RCNN pipeline used to identify the objects which are given in the casing. It includes the maps which are separated by a spine when it begins. The strategy of CNN, for the purpose of district proposition network in order to measure the jump boxes, is utilized. Secondly, it contains the ROI layers of pooling which mixes the extricated highlighted maps with the proposition of the juamping box and also empowers the classifier in order to yield the both of the class of article and a suitable bounding box. The RPN part is also very significant and it is used to make the bouncing box recommendation as it slides little n x n window and further plan these items into a lower-dimensional vector which is highlighted [30]. The specific type of vector is protected by the two connected layers of equal size and further goes about a crate layer of repressor in order to encode the focus of directions of the concerned bouncing box, its width, tallness, and the order of case layer in order to demonstrate the case which is consisted of important items. Besides this the sliding window is used to create certain kind of recommendations, secures the call, used to represent few scales and the proportions, and the totaling k = 9 for the each space in area. The reg and cls layers are consisted of 4k and 2k components which are separated and for the each element in the map there used to W x H x k recommendations. In this equation W x H is compared to the sized map. It is important to note that during the first execution the window having a size of n=3 can be seen and the reg and cls categories of the layers are equally shared in the spatial localities which are dissected through the sliding window in order to hold the exhibition of further advancements. In Fig. 8 the Faster R-CNN pipeline is summarized and employed to detect the concerned objects inside the frame which is given. One direction is used as the input of RPN to generate candidate regions, and the other continues to arrive at the unique convolution layer to generate a higher-dimensional feature map. In the practical application of RPN, the fault area range of insulators is obtained by inputting the characteristic graph of the fault node in the sliding window area [31]. Specifically, starting from the feature maps extracted by a backbone CNN, this method first employs a region proposal network to estimate bounding boxes (i.e., proposed regions) and whether or not a specific region contains a relevant object. Secondly, it also implements an ROI pooling layer which is used to merge the extracted feature maps having bounding boxes proposals. It enables a classifier to output the object class and an appropriate bounding box containing it [32].

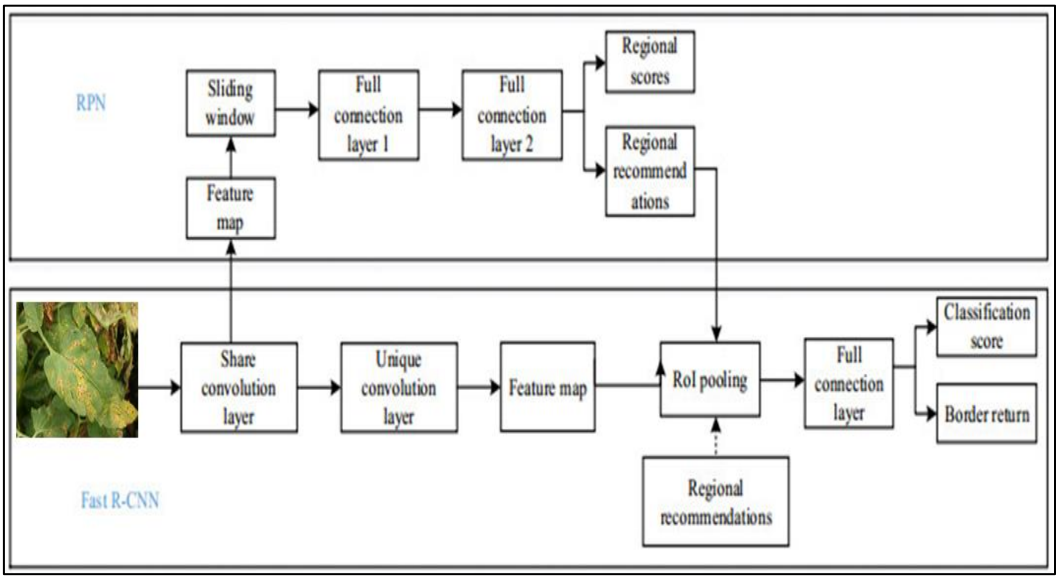

Fig. 8. Faster R-CNN RPN and ROI pooling scheme

Looking at the RPN component we know that it is used to create bounding boxes proposals by moving a small n×n window on the extracted features and also used to map these items into a lower-dimensional. This concerned vector is used to fed to two parallel fully-connected layers which act as a box-regressor layer (i.e., reg), in order to encode the bounding boxes center coordinates, its width and height, and classification layer in order to indicate that a box which is consisted of relevant object. Furthermore, the sliding window generates proposals called anchors, accounting for the several types of the scales and ratios, and the totaling rate of k=9 proposals. The 4k and 2k elements are contained in the layers of reg and cs\ls. For all the featured maps the W x H x k is proposed [33].In this equation W x H is represents the size of map. It is important to note that the implementation of original the size of the sliding window is n=3 and in order to maintain the improved performances clas and reg layers are properly shared across the all predefined directions which is done by the sliding window. Fig. 9 regarding the ROI pooling and final object detection which starts from the featured maps is extracted by the CNN backbone. The proposals are generally computed through the RPN which is an adaptive pooling layer and applied in order to accurately merge the twin inputs into the single vector [34]. Subsequently, the pooled which are inputs are analyzed by the layers which are fully connected. Its output is fed to two similar classifiers which are used to obtain the final bounding box and object category of prediction for the purpose of input frame, respectively [35].

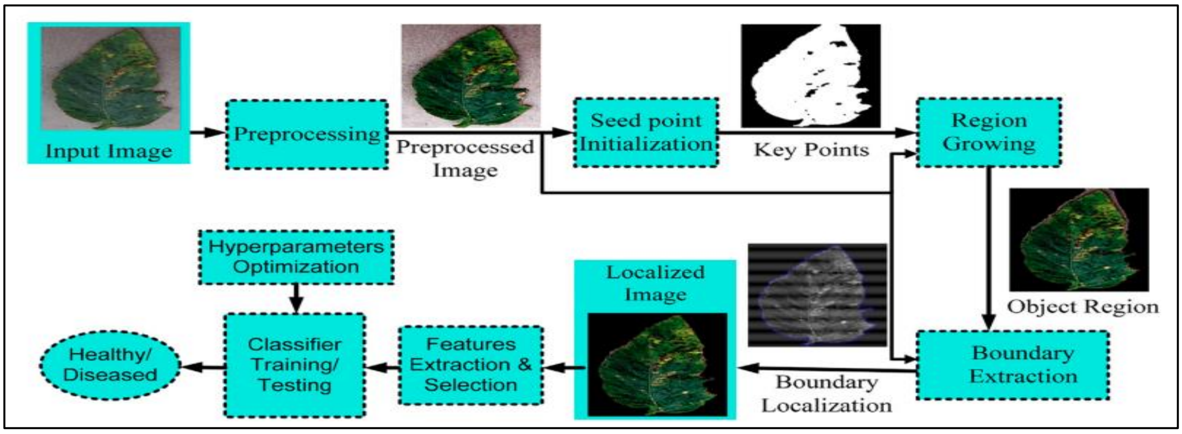

Fig. 9. Flowchart of Image Localization and Classification Process

3.9. Multi-Stream Faster R-CNN Loss Functions

In accordance with [36], the presented methodology could be trained into an end-to-end fashion; since, in this work relevant modifications were only applied to the backbone CNN which used in order to get the extracted features from an image given as input. More accurately, Faster R-CNN pipeline employs a multi-task loss function associated with the bounding box regression and object classification tasks. Formally, as per the definition [37], for a given mini-batch, the function to be minimized is computed in accordance to the equation given below:

$$L(\{P_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}\,(Pi, Pi\,*) + \lambda l\frac{1}{N_{reg}}\sum_i Pi^{*}L_{reg}(t_i,t_i) \tag{2}$$

In this equation i indicates the i-th anchor of the mini-batch and pi and p∗i are representing the predicted and genuine truth probability of the anchor which is associated with a relevant object. Further the ti and t∗i are representing the generated and truth vectors which are consisted of the parameterized bounding box coordinates. Ncls and Nreg correspond to normalization terms based on the batch size and the number of proposed anchors, respectively, while λl is a balancing parameter to ensure both losses have similar weights. Moreover, Lcls is a binary cross-entropy loss function, while Lreg is a regression loss using the robust function defined [38].

$$L_{reg}(t_i,t_i) = \sum_{i\in\{x,y,w,h\}} smooth_{L1}\,(t_i,t_i) \tag{3}$$

ISSN 2086-8138 (print) | 2745-7249 (online)
Vol. 17., No. 1, January 2023, pp. 39-56

Where the smooth function is computed as follows:

$$smooth_{L1}(x) = \begin{cases} 0.5x^2 \; if \; |x| < 1; \\ |x| - 0.5 \; otherwise \end{cases} \tag{4}$$

Finally, concerning the bounding box regression, we have employed the same parameterization defined in [39], described via the following equations:

$$\begin{aligned} t_x &= (x - x_a)/w_a, \quad t_y = (y - y_a)/h_a, \\ t_w &= log(w/w_a), \quad t_h = log(h/h_a), \\ t_x^* &= (x^* - x_a)/w_a, \quad t_y^* = (y^* - y_a)/h_a, \\ t_w^* &= log(w^*/w_a), \quad t_h^* = log(h^*/h_a), \end{aligned} \tag{5}$$

In the above given equation x, y, w, and h are corresponding to the coordinating bounding box center; the width and the height respectively. While the other variables x, xa, and x∗ are associated with the futuristic prediction of the bounding box, the proposed anchor bounding box, and the ground truth bounding box. The same reasoning applies to the other parameters (i.e., y, w, and h) [39]–[41].

4. Evaluation Metrics

Various boundaries are assessed while using the proposed method and specific among those are: the Precision, Recall, Accuracy, Detection Rate, the area under bend, and F1 score. Detection Rate: The detection rate can be measured using the formula available [42].

$$\text{Detection Rate} = \frac{TP}{(TP+FN)} \tag{7}$$

AUC: The AUC can be measured using the formula available.

$$\text{AUC} = \frac{1}{2}\left(\frac{TP}{(TP+FN)} + \left(\frac{TN}{(TN+FP)}\right)\right) \tag{8}$$

Accuracy: The accuracy can be measured using the formula available.

$$\text{Aaccuracy} = \frac{(TP+TN)}{(TP + TN + FP + FN)} \tag{9}$$

Precision: The precision can be measured using the formula available.

$$\text{Precision} = \frac{TP}{(\text{TP} + \text{FP})} \tag{10}$$

FI Score: The F1 score can be measured using the formula available.

$$\text{F1 Score} = \frac{2TP}{(2\text{TP} + \text{FP} + \text{FN})} \text{ [43]-[46].} \tag{11}$$

A confusion matrix is used to check the outcome of the classification. Accuracy can be measured with the help of evolution matrix.

5. Tools and Technology

To perform experiments on the computer with Jupyter Notebook (Anaconda3), Intel processor 3.91 GHz, 8 GB RAM, and Core i3 minimum 5th generation. The sample of affected tomato images from datasets was used as input images. Python programming language is used to perform this experiment. Different programming libraries would be used to arrange the tomato images. These libraries extract useful features and arrange them into metrics to evaluate the proposed technique.

6. Results and Discussions

Tomato leaf disease detection has effectively been done through different algorithms in existing research works. A group of researchers uses different deep learning techniques. The exploration proposed Faster RCNN for such a reason and tested the proposed method by using an ongoing dataset. Study sorts out region under bend, exactness, location rate, accuracy, and F1 score. Analysis shows magnificent results against all evaluation grids which shows that the proposed method is better contrasted with as of late used for tomato diseases. Initially, training framework by utilizing ongoing images information and preparing the framework shows generally excellent outcomes. Train framework on the information of 80% tomato images of the chosen dataset to upgrade the estimation of the appraisal grids. In the wake of preparation, test the framework with 20% images of the tomato which are not utilized in the preparation of the framework. After the examination, contrast the outcomes and the after effects of recently utilized strategies for recognition of tomato leaf diseases. The improvement in disease detection is shown which demonstrates that the procedure is very better compared to past methods. Table 1 present result of comparison between proposed and existing techniques.

Table 1. Comparison between Proposed and Existing Techniques

Existing Work	System	Train/Test/ Validation ratio	Tomato Leaf Diseases Detection Accuracy %	Tomato Leaf Diseases Detection Precision %
[10]	GPU	80/10/10	75.37%	-
[19]	GPU	60/20/20	80.95%	-
[23]	GPU	70/20/10	60.92%	73.07%
[13]	-	-	90.0%	-
[20]	-	70/30	96.73%	-
Proposed work	CPU	80/20	98.6%	91.0%

Table 1 the comparison results of the proposed model and existing techniques as it looks at the after effects of the proposed approach for all assessment parameters described above. The above table presents the outcomes of the technique proposed in this study for each assessment parameter. Faster RCNN tomato leaf disease detections are more proficient and adequate. F1 score is a general assessment of the framework as it were of exactness and accuracy. A framework that is more exact and exact has more F1 scores and is known as the F1 measure. The F1 score for the proposed arrangement is more since the framework is more exact and exact. Detection rate (DR) is the hitting pace of the framework on each heartbeat implies it distinguishes a beat or not. As true positive TP, true negative TN, False positive FP, and False Negative FN. The framework recognizes the hit rate in any of these qualities. A framework with a high detection rate is more powerful. Fig. 10 present the result of Comparison between proposed and existing techniques.

Jurnal Informatika
Vol. 17., No. 1, January 2023, pp. 39-56

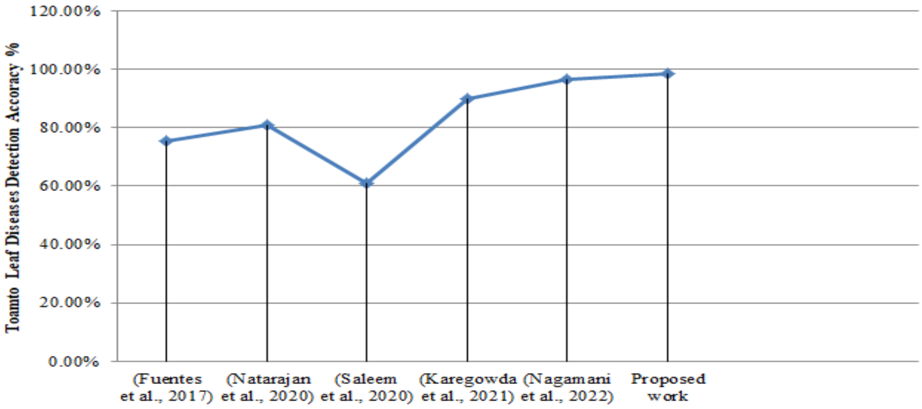

Fig. 10. Comparison between Proposed and Existing Techniques

The degree of accuracy denotes how well the structure acknowledges the disease. Because of its high precision value, the system is more precise and unnecessarily close to the best situation. Precision is determined by the number of authentic positive and counterfeit positive data attempts. Faster RCNN based tomato leaf diseases detection is more accurate than those that are currently in use. It is much more similar to the best basis for evaluating tomato leaf diseases. Fig. 11 shows that accuracy of the proposed strategy when contrasted by the current strategies.

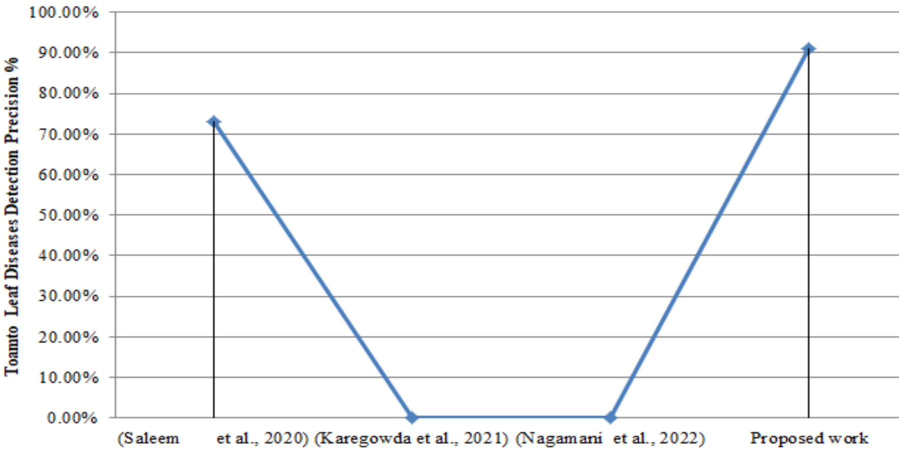

Fig. 11. Precision of Compared Techniques

Precision is the detection of a true positive. This implies as numerous frameworks detect as true positive. Outcomes show that the present technique is exact in detecting tomato leaf diseases. Fig. 12 shows the precision of compared techniques.

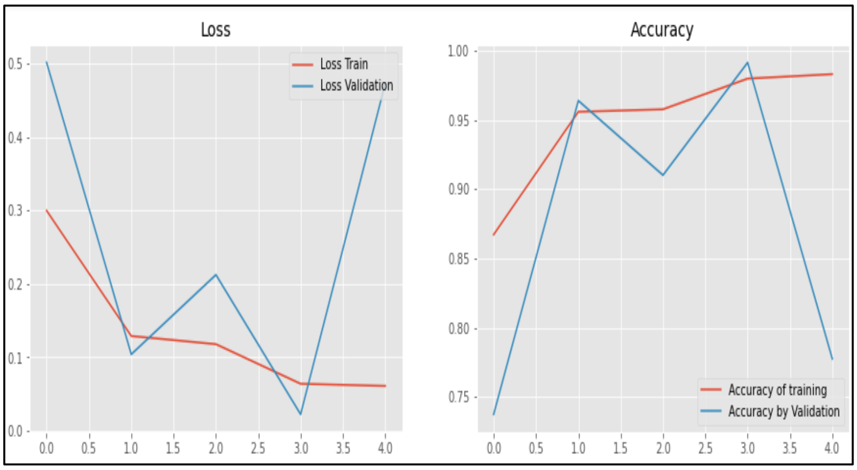

Fig. 12. .Training and Validation Losses and Accuracies for Faster RCNN

Fig. 12 described that training and validation accuracy curves for the Faster RCNN technique show a continuous increase in the training accuracy and a decrease in the validation accuracy. These results depict the information related to the loss and accuracy generated during model implementation. It can be said that the loss values obtained from the model reveal that with the increase in epochs count, it reported a decrease in both training and validation loss. However, the validation loss does not have a constant decrease count and therefore represents increasing values over certain epoch's iterations. This commonly happens because the loss count during implementation has to be reduced for increasing the accuracy of the model. They also observed the accuracy count in a way that the constant increasing trend has been observed for the training accuracy in comparison to the results retrieved for the validation accuracy. To this end, it can be said that more suitable values have been obtained in the training case as compared to the results of the validation case. Fig. 6 shows that the proposed technique has training and validation losses and accuracy in tomato leaf detection using Faster RCNN.

7. Conclusion

This paper discusses the different diseases in tomato plants. There are many techniques already developed to detect tomato leaf diseases which have some issues in the detection of tomato leaf diseases. This research used a deep learning technique to detect tomato leaf diseases. The model, called multi-Stream Faster R-CNN, is made by a multi-stream CNN as a spine, and by the norm RPN of the Faster R-CNN. The spine utilizes a pyramidal methodology, i.e., various streams with various portion sizes, to separate features at various scales, taking into consideration effectively recognizing objects at various flight statures. Analysis shows that the proposed technique is more effective than the previously designed methodology. The study also focuses on the privacy of the dataset. Limitations the purposed method mainly focuses on the three major types of tomato leaf disease detection. But this proposed method can be applied to other plants such as flowers, roots, and others. Future Directions and Research Opportunities Faster RCNN based method has been further improved and analyze the detection of tomato leaf diseases and various parts of other plants. Decrease data preprocessing and the size of model for compact devices. Finally, the dataset used in this result is very limited for better performance it's suggested to increase the more images of different tomato plant diseases.

References

[1] A. Abbas, S. Jain, M. Gour, and S. Vankudothu, "Tomato plant disease detection using transfer learning with C-GAN synthetic images," *Comput. Electron. Agric.*, vol. 187, p. 106279, Aug. 2021, doi: 10.1016/J.COMPAG.2021.106279.

[2] M. Agarwal, A. Singh, S. Arjaria, A. Sinha, and S. Gupta, "ToLeD: Tomato Leaf Disease Detection using Convolution Neural Network," *Procedia Comput. Sci.*, vol. 167, pp. 293–301, Jan. 2020, doi: 10.1016/J.PROCS.2020.03.225.

[3] S. Ashok, G. Kishore, V. Rajesh, S. Suchitra, S. G. G. Sophia, and B. Pavithra, "Tomato Leaf Disease Detection Using Deep Learning Techniques," pp. 979–983, Jul. 2020, doi: 10.1109/ICCES48766.2020.9137986.

[4] Y. Altuntaş, Y. Altuntaş, and A. Fatih Kocamaz, "Deep Feature Extraction for Detection of Tomato Plant Diseases and Pests based on Leaf Images," *Celal Bayar Univ. J. Sci.*, vol. 17, no. 2, pp. 145–157, Jun. 2021, doi: 10.18466/CBAYARFBE.812375.

[5] A. Bhujel, N. E. Kim, E. Arulmozhi, J. K. Basak, and H. T. Kim, "A Lightweight Attention-Based Convolutional Neural Networks for Tomato Leaf Disease Classification," *Agric. 2022, Vol. 12, Page 228*, vol. 12, no. 2, p. 228, Feb. 2022, doi: 10.3390/AGRICULTURE12020228.

[6] J. Bhuvana and T. T. Mirnalinee, "An approach to Plant Disease Detection using Deep Learning Techniques," *Iteckne*, vol. 18, no. 2, pp. 161–169, 2021, doi: 10.15332/iteckne.v18i2.2615.

[7] E. Cengil and A. Çınar, "Hybrid convolutional neural network based classification of bacterial, viral, and fungal diseases on tomato leaf images," *Concurr. Comput. Pract. Exp.*, vol. 34, no. 4, p. e6617, Feb. 2022, doi: 10.1002/CPE.6617.

[8] B. Cheng and E. T. Matson, "A feature-based machine learning agent for automatic rice and weed discrimination," *Lect. Notes Artif. Intell. (Subseries Lect. Notes Comput. Sci.*, vol. 9119, pp. 517–527, 2015, doi: 10.1007/978-3-319-19324-3_46.

[9] V. Eligar, U. Patil, and U. Mudenagudi, "Performance Analysis of Deep Learning Algorithms Toward Disease Detection: Tomato and Potato Plant as Use-Cases," *Smart Innov. Syst. Technol.*, vol. 286, pp. 595–606, 2022, doi: 10.1007/978-981-16-9873-6_54.

[10] A. Fuentes, S. Yoon, S. C. Kim, and D. S. Park, "A Robust Deep-Learning-Based Detector for Real-Time Tomato Plant Diseases and Pests Recognition," *Sensors 2017, Vol. 17, Page 2022*, vol. 17, no. 9, p. 2022, Sep. 2017, doi: 10.3390/S17092022.

[11] C. Hu, X. Liu, Z. Pan, and P. Li, "Automatic detection of single ripe tomato on plant combining faster R-CNN and intuitionistic fuzzy set," *IEEE Access*, vol. 7, pp. 154683–154696, 2019, doi: 10.1109/ACCESS.2019.2949343.

[12] H. Jain, A. Vikram, Mohana, A. Kashyap, and A. Jain, "Weapon Detection using Artificial Intelligence and Deep Learning for Security Applications," *Proc. Int. Conf. Electron. Sustain. Commun. Syst. ICESC 2020*, pp. 193–198, Jul. 2020, doi: 10.1109/ICESC48915.2020.9155832.

[13] A. G. Karegowda, R. Jain, and G. Devika, "State-of-Art Deep Learning Based Tomato Leaf Disease Detection," *Proc. 3rd Int. Conf. Integr. Intell. Comput. Commun. Secur. (ICIIC 2021)*, vol. 4, pp. 303–311, Sep. 2021, doi: 10.2991/AHIS.K.210913.038.

[14] T. L. Lin, H. Y. Chang, and K. H. Chen, "The pest and disease identification in the growth of sweet peppers using faster r-cnN and mask r-CNn," *J. Internet Technol.*, vol. 21, no. 2, pp. 605–614, 2020, doi: 10.3966/160792642020032102027.

[15] J. Liu and X. Wang, "Early recognition of tomato gray leaf spot disease based on MobileNetv2-YOLOv3 model," *Plant Methods*, vol. 16, no. 1, pp. 1–16, Jun. 2020, doi: 10.1186/S13007-020-00624-2.

[16] B. Luna-Benoso, J. C. Martínez-Perales, J. Cortés-Galicia, R. Flores-Carapia, and V. M. Silva-García, "Detection of Diseases in Tomato Leaves by Color Analysis," *Electron. 2021, Vol. 10, Page 1055*, vol. 10, no. 9, p. 1055, Apr. 2021, doi: 10.3390/ELECTRONICS10091055.

[17] S. V. Militante and B. D. Gerardo, "Detecting Sugarcane Diseases through Adaptive Deep Learning Models of Convolutional Neural Network," *ICETAS 2019 - 2019 6th IEEE Int. Conf. Eng. Technol. Appl. Sci.*, Dec. 2019, doi: 10.1109/ICETAS48360.2019.9117332.

[18] Y. Mu, T.-S. Chen, S. Ninomiya, and W. Guo, "Intact Detection of Highly Occluded Immature Tomatoes on Plants Using Deep Learning Techniques," *Sensors*, vol. 20, no. 10, p. 2984, May 2020, doi: 10.3390/s20102984.

[19] V. A. Natarajan, M. Macha Babitha, and M. Sunil Kumar, "Detection of disease in tomato plant using Deep Learning Techniques," *Int. J. Mod. Agric.*, vol. 9, no. 4, pp. 525–540, Dec. 2020, Available: http://modern-journals.com/index.php/ijma/article/view/374

[20] H. S. Nagamani and H. Sarojadevi, "Tomato Leaf Disease Detection using Deep Learning Techniques," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 1, pp. 305–311, 2022, doi: 10.14569/IJACSA.2022.0130138.

[21] T. A. Salih, A. J. Ali, and M. N. Ahmed, "Deep Learning Convolution Neural Network to Detect and Classify Tomato Plant Leaf Diseases," *OALib*, vol. 07, no. 05, pp. 1–12, 2020, doi: 10.4236/OALIB.1106296.

[22] S. Santhosh Kumar and B. K. Raghavendra, "Diseases Detection of Various Plant Leaf Using Image Processing Techniques: A Review," *2019 5th Int. Conf. Adv. Comput. Commun. Syst. ICACCS 2019*, pp. 313–316, Mar. 2019, doi: 10.1109/ICACCS.2019.8728325.

[23] M. Hammad Saleem, S. Khanchi, J. Potgieter, and K. Mahmood Arif, "Image-Based Plant Disease Identification by Deep Learning Meta-Architectures," *Plants 2020, Vol. 9, Page 1451*, vol. 9, no. 11, p. 1451, Oct. 2020, doi: 10.3390/PLANTS9111451.

[24] E. Suryawati, R. Sustika, R. S. Yuwana, A. Subekti, and H. F. Pardede, "Deep structured convolutional neural network for tomato diseases detection," *2018 Int. Conf. Adv. Comput. Sci. Inf. Syst. ICACSIS 2018*, pp. 385–390, Jan. 2019, doi: 10.1109/ICACSIS.2018.8618169.

[25] Q. Wang and F. Qi, "Tomato diseases recognition based on faster RCNN," *Proc. - 10th Int. Conf. Inf. Technol. Med. Educ. ITME 2019*, pp. 772–776, Aug. 2019, doi: 10.1109/ITME.2019.00176.

[26] Q. Wang, F. Qi, M. Sun, J. Qu, and J. Xue, "Identification of Tomato Disease Types and Detection of Infected Areas Based on Deep Convolutional Neural Networks and Object Detection Techniques," *Comput. Intell. Neurosci.*, vol. 2019, 2019, doi: 10.1155/2019/9142753.

[27] K. Yang and Z. Song, "Deep Learning-Based Object Detection Improvement for Fine-Grained Birds," *IEEE Access*, vol. 9, pp. 67901–67915, 2021, doi: 10.1109/ACCESS.2021.3076429.

[28] S. Zhao, Y. Peng, J. Liu, and S. Wu, "Tomato Leaf Disease Diagnosis Based on Improved Convolution Neural Network by Attention Module," *Agric. 2021, Vol. 11, Page 651*, vol. 11, no. 7, p. 651, Jul. 2021, doi: 10.3390/AGRICULTURE11070651.

[29] J. Zhong, Z. Liu, Z. Han, Y. Han, and W. Zhang, "A CNN-Based Defect Inspection Method for Catenary Split Pins in High-Speed Railway," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 8, pp. 2849–2860, Aug. 2019, doi: 10.1109/TIM.2018.2871353.

[30] U. Adhikari, A. P. Nejadhashemi, M. R. Herman, and J. P. Messina, "Multiscale Assessment of the Impacts of Climate Change on Water Resources in Tanzania," *J. Hydrol. Eng.*, vol. 22, no. 2, p. 05016034, Feb. 2017, doi: 10.1061/(ASCE)HE.1943-5584.0001467.

[31] B. C. Schumacher *et al.*, "Design alternatives for internationally distributed learning factories in global production engineering," *Procedia Manuf.*, vol. 45, pp. 392–397, 2020, doi: 10.1016/j.promfg.2020.04.042.

[32] C.-C. Tsai, "Biological and bioinspired photonic materials: From butterfly wings and silk fibers to radiative-cooling textiles and object-recognition smart glass," 2022, doi: 10.7916/D8-HGEZ-1R37.

[33] M. E. Elhamahmy, H. N. Elmahdy, and I. A. Saroit, "A New Approach for Evaluating Intrusion Detection System," *Artif. Intell. Syst. Mach. Learn.*, vol. 2, no. 11, pp. 290–298, 2010, Available: http://www.ciitresearch.org/dl/index.php/aiml/article/view/AIML112010001

[34] M. Bekkar and T. A. Alitouche, "Imbalanced Data Learning Approaches Review," *Int. J. Data Min. Knowl. Manag. Process*, vol. 3, no. 4, pp. 15–33, 2013, doi: 10.5121/ijdkp.2013.3402.

[35] H. Yoshino *et al.*, "A predictive threshold value for the diagnosis of early pregnancy in cows using interferon-stimulated genes in granulocytes," *Theriogenology*, vol. 107, pp. 188–193, Feb. 2018, doi: 10.1016/J.THERIOGENOLOGY.2017.11.014.

[36] A. S. Hussein, R. S. Khairy, S. M. Mohamed Najeeb, and H. T. Salim ALRikabi, "Credit Card Fraud Detection Using Fuzzy Rough Nearest Neighbor and Sequential Minimal Optimization with Logistic Regression," *Int. J. Interact. Mob. Technol.*, vol. 15, no. 05, pp. 24–42, Mar. 2021, doi: 10.3991/IJIM.V15I05.17173.

[37] F. M. M. Mokbal, W. Dan, A. Imran, L. Jiuchuan, F. Akhtar, and W. Xiaoxi, "MLPXSS: An Integrated XSS-Based Attack Detection Scheme in Web Applications Using Multilayer Perceptron Technique," *IEEE Access*, vol. 7, pp. 100567–100580, 2019, doi: 10.1109/ACCESS.2019.2927417.

[38] S. Dalabasmaz, D. Dittrich, I. Kellner, T. Drewello, and M. Pischetsrieder, "Identification of peptides reflecting the storage of UHT milk by MALDI-TOF-MS peptide profiling," *J. Proteomics*, vol. 207, p. 103444, Sep. 2019, doi: 10.1016/J.JPROT.2019.103444.

[39] P. Kumar and R. S. Thakur, "Liver disorder detection using variable- neighbor weighted fuzzy K nearest neighbor approach," *Multimed. Tools Appl.*, vol. 80, no. 11, pp. 16515–16535, May 2021, doi: 10.1007/S11042-019-07978-3.

[40] L. J. Latecki, A. Lazarevic, and D. Pokrajac, "Outlier detection with kernel density functions," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 4571 LNAI, pp. 61–75, 2007, doi: 10.1007/978-3-540-73499-4_6.

[41] R. Thangaraj, S. Anandamurugan, P. Pandiyan, and V. K. Kaliappan, "Artificial intelligence in tomato leaf disease detection: a comprehensive review and discussion," *J. Plant Dis. Prot.*, vol. 129, no. 3, pp. 469–488, Jun. 2022, doi: 10.1007/S41348-021-00500-8.

[42] S. Nandhini and K. Ashokkumar, "Improved crossover based monarch butterfly optimization for tomato leaf disease classification using convolutional neural network," *Multimed. Tools Appl.*, vol. 80, no. 12, pp. 18583–18610, May 2021, doi: 10.1007/S11042-021-10599-4.

[43] X. Chen, G. Zhou, A. Chen, J. Yi, W. Zhang, and Y. Hu, "Identification of tomato leaf diseases based on combination of ABCK-BWTR and B-ARNet," *Comput. Electron. Agric.*, vol. 178, p. 105730, Nov. 2020, doi: 10.1016/J.COMPAG.2020.105730.

[44] A. Ullah, H. Aznaoui, C. B. Sahin, I. Daanoune, and O. B. Dinle, "CloudIoT paradigm acceptance for e-learning: analysis and future challenges," *J. Inform.*, vol. 16, no. 3, pp. 151–165, Sep. 2022, doi: 10.26555/JIFO.V16I3.A21744.

[45] A. Ullah, N. Mohd Nawi, A. Arifianto, I. Ahmed, M. Aamir, and S. N. Khan, "Real-time wheat classification system for selective herbicides using broad wheat estimation in deep neural network," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 9, no. 1, pp. 153–158, 2019, doi: 10.18517/IJASEIT.9.1.5031.

[46] A. Ullah, I. Laassar, C. B. Şahin, O. B. Dinle, and H. Aznaoui, "Cloud and internet-of-things secure integration along with security concerns," *Int. J. Informatics Commun. Technol.*, vol. 12, no. 1, pp. 62–71, Apr. 2023, doi: 10.11591/IJICT.V12I1.PP62-71.