IMPROVED YOLOV4 MODEL AND MEDIAN FILTER BASED MAJOR DISEASES DETECTION IN TURMERIC PLANT

Mrs.R.Thilakavathi Research Scholar, Park's College, Chinnakkarai.

Dr.S.Nithya Assistant Professor, Department of Computer Science, Park's College, Chinnakarai, Tirupur, Tamilnadu, India.

ABSTRACT

Major disease causing micro-organisms are bacteria and viruses, which are not visible when it affects the plant at initial stage. The human naked eyes after only the later stage it's visible and its affected whole parts of plants. Artificial Intelligence is an emerging sector in all fields of works for automation and to improve efficiency. It also included in agricultural sector to improve crop yield by identify the disease affection at early and classify type of disease affected for taking precaution measurements to prevent spreading to other plants in field. In existing work to enhance the detection accuracy of turmeric diseases, a deep learning-based technique called the Improved YOLOV3-Tiny model is proposed. However input images have more noise and it will affect the classifier performance. Existing work does not focused on this issue. Improved YOLOV3-Tiny requires a significant amount of memory to run, which can be a challenge for devices with limited resources. To overcome this issue in this work introduced an improved model for major diseases detection in turmeric plant. In this work initially data augmentation is performed to improve the classifier accuracy using methods such as image rotation, image colour, image brightness transformation, motion blur transformation and Cycle-GAN deep learning model. Preprocessing is computed using median filtering based noise removal. After that texture features are extracted using Gray Level Co-Occurrence Matrix (GLCM) method. Major diseases in turmeric plants are detected using Improved YOLOV4 (IYOLOV4) model. Results show the effectiveness of the proposed model in terms of precision, recall, accuracy and f-measure.

Keywords: Turmeric plant, Diseases detection, Data augmentation, median filtering, texture features and Improved Yolov4.

1. INTRODUCTION

Agricultural is the backbone of country and which the basic term of food production for human. The production of agricultural land decreases yearly due to different types disease affection. Turmeric plant production materials are play in our daily life routine on our food preparation, health care etc,[1]. The major turmeric plant affecting diseases are Leaf Spot, Leaf Blotch which are air borne diseases and destroy whole agricultural at very short time. For instant, leaf spot is affected badly in rainy seasons under humid conditions. It causes damage to a greater extent by reducing rhizome size up to 52 percent [2]. The infection of such diseases causes the change in the color and appearance of the turmeric leaves [3]. The existing methods for plant disease detection simply by naked eye observation or laboratory based techniques by experts is time consuming and requires continuous monitoring of plant. Therefore, if the plant monitoring methods can be stored by using some programming language into an automatic module then the process can be error tolerant. Hence image processing plays a vital role in disease detection and analysis [4]. In the recent works several image processing methods and algorithms are used for the detection of turmeric plant diseases. These machine learning models have been applied to a variety of plant species, including turmeric [5,6]. Kuricheti and Supriya used a k-means algorithm for image segmentation and a support vector machine (SVM) classifier for feature extraction to develop a GUI system for detecting turmeric diseases and controlling their spread. In another research, CNN based on the VGG16 planning for detecting turmeric sicknesses at an initial stage and protecting from fast sickness spread. However,

these methods do not produce the sufficient accuracy results. To avoid this problems in existing work to enhance the detection accuracy of turmeric diseases, a deep learning-based technique called the Improved YOLOV3-Tiny model is proposed. However input images have more noise and it will affect the classifier performance. Existing work does not focused on this issue. Improved YOLOV3-Tiny requires a significant amount of memory to run, which can be a challenge for devices with limited resources. To overcome this issue in this work introduced an improved model for major diseases detection in turmeric plant. In this work initially data augmentation is performed to improve the classifier accuracy using methods such as image rotation, image colour, image brightness transformation, motion blur transformation and Cycle-GAN deep learning model. Preprocessing is computed using median filtering based noise removal. After that texture features are extracted using Gray Level Co-Occurrence Matrix (GLCM) method. Major diseases in turmeric plants are detected using Improved YOLOV4 (IYOLOV4) model.

2. RELATED WORKS

Suryawanshi and Khurjekar[2021][7]Proposed a system which aims to develop an auto-guided drone that can take the images of crop leaves as input. These images will then be processed by applying Convolutional Neural Network (CNN) to detect the diseases which are affecting the crops. This system will also help mark the most affected regions of fields. By using this system,can increase the productivity of the crop. Kuricheti and Supriya[2019][8]developed an algorithm for detecting and preventing the spreading of diseases to the whole crop and results in high quality crop production. The data base of different leaf images was created and processed using k-Means image segmentation and leaf images textural analysis was carried out using GLCM. SVM classifier is used to classify the feature extracted images after ranking their attributes using an information gain algorithm. A GUI has been created to portray the various stages of the image processing algorithm and detect the two leaf diseases. Albattah, et al [2022][9]introduced a robust plant disease classification system by introducing a Custom CenterNet framework with DenseNet-77 as a base network. The presented method follows three steps. In the first step, annotations are developed to get the region of interest. Secondly, an improved CenterNet is introduced in which DenseNet-77 is proposed for deep keypoints extraction. Finally, the one-stage detector CenterNet is used to detect and categorize several plant diseases. To conduct the performance analysis, we have used the PlantVillage Kaggle database, which is the standard dataset for plant diseases and challenges in terms of intensity variations, color changes, and differences found in the shapes and sizes of leaves. Both the qualitative and quantitative analysis confirms that the presented method is more proficient and reliable to identify and classify plant diseases than other latest approaches. Tulshanand Raul[2019][10]applied a plant leaf disease detection technique to detect a disease from the input images. This process involved steps like image pre-processing, image segmentation, feature extraction. Furthur K Nearest Neighbor (KNN) classification is applied on the outcome of these three stages. Proposed implementation has shown 98.56% of accuracy in predicting plant leaf diseases. It also presents other information regarding a plant leaf disease that is Affected Area, Disease Name, Total Accuracy, Sensitivity and Elapsed Time. Ramesh et al [2018][11]used random Forest in identifying between healthy and diseased leaf from the data sets created. Our proposed paper includes various phases of implementation namely dataset creation, feature extraction, training the classifier and classification. The created datasets of diseased and healthy leaves are collectively trained under Random Forest to classify the diseased and healthy images. For extracting features of an image we use Histogram of an Oriented Gradient (HOG). Overall, using machine learning to train the large data sets available publicly gives us a clear way to detect the disease present in plants in a colossal scale. Akanksha et al [2021][12]developed an efficient automated diagnosis of maize plants. The proposed methodology is consists of four stages namely, preprocessing, feature extraction, classification, and segmentation. Initially, the images are converted into RGB format and the images present in the noises are removed. Then, the R band is given to the feature extraction

stage. Then, the selected attributes are fed to the classifier to classify an image as normal or abnormal. For classification, an optimized probabilistic neural network (OPNN) is utilized. The PNN classifier is improved by using artificial jelly optimization (AJO) algorithm. Finally, the Northern leaf blight disease leaf images are fed to the segmentation stage to separate the affected portion of a leaf. The effectiveness of the proposed lead disease classification is investigated base on the various quality metrics namely, accuracy, sensitivity, and specificity. Tm et al [2018][13]aimed to propose a work to find a solution to the problem of tomato leaf disease detection using the simplest approach while making use of minimal computing resources to achieve results comparable to state of the art techniques. Neural network models employ automatic feature extraction to aid in the classification of the input image into respective disease classes. This proposed system has achieved an average accuracy of 94-95 % indicating the feasibility of the neural network approach even under unfavourable conditions.

3. PROPOSED METHODOLOGY

This section discusses the proposed model in detail. Proposed model consist of four phases first one is image augmentation using image rotation, image colour, image brightness transformation, motion blur transformation and Cycle-GAN deep learning model, second one is image pre-processing using median filtering, third one is feature extraction using Gray Level Co-Occurrence Matrix (GLCM) and fourth one is Classification using Improved Yolov4.Overall architecture of the proposed model is shown in figure 1.

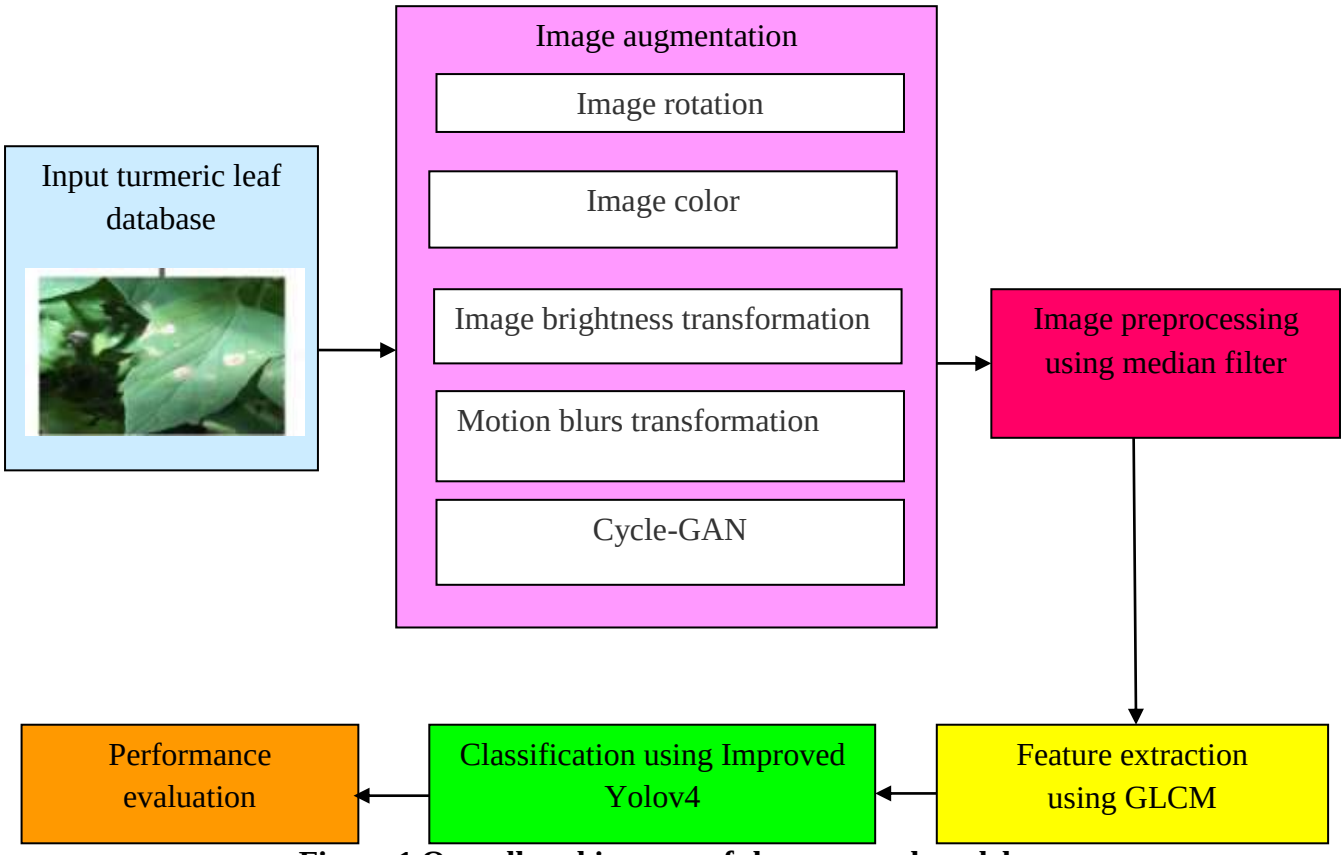

Figure 1.Overall architecture of the proposed model

3.1. Data augmentation

Deep learning methods need more datasets for accurate object detection. Huge dataset could improve the object detection accuracy rate and increase performance. In this work, 1,600 images are taken from a real-time environment. In order to expand the training image dataset, data augmentation is applied to artificially enlarge the dataset. Network performance can be improved by using data

augmentation methods such as brightness transformation, image rotation, and histogram equalization. In this work, traditional augmentation techniques like image rotation, image colour, image brightness transformation, and motion blur transformation are used. CycleGAN, the deep learning model for data augmentation is also used for finding better detection accuracy. The various augmentation methods used in this work are discussed as follows:

3.1.1. Image rotation

Actual pictures are rotated 90° and 180° to expand the dataset. Mirroring of images is also done. The neural network performance can be improved by using a larger dataset with image rotation.

3.1.2. Image color

The human vision ability to detect the color invariance of the leaves under changing light conditions, but imaging technology lacks this capability. Various lighting circumstances can result in a specific image and true color divergence. To reduce the effect of light on color rendering, the Gray world method is utilized. This approach is based on the gray-world hypothesis, which states that an image with a significant number of color changes will have the same grey value if the average value of the components R, G, and B is the same. The grey world method implies that the average light reflection from objects is normally a fixed value, similar to grey. In the training set for color invariance, this color balance algorithm is applied to images.

3.1.3. Image brightness transformation

The brightness transformation is a typical way of data augmentation used to improve the robustness of a network. Multiplying the proportionate coefficient near 1.0 with the original RGB image can result in increased or decreased image brightness.

3.1.4. Motion blurs transformation

The distanced camera can make incorrect focusing and movement of the camera may lead to blur image. To obtain the blur image transformation, equations (1) and (2) are used in this study. Parameters L (length reflects the linear motion of the camera pixels) and q (theta is the angle between horizontal line and direction of the motion of the camera) are set based on the application type.

$$g(x,y) = h(x,y) * f(x,y) \quad (1)$$

$$h(x,y) = \{\frac{1}{L}, \sqrt{x^2+y^2} \leq L \text{ and } \frac{y}{x} = tan\theta \;\; 0, \quad otherwiswe \quad (2)$$

where: g(x;y) is a motion-blurred image, h(x;y) is a degenerate function and f (x;y) is the original image. In order to prevent over-fitting in deep learning algorithms, positive images with inadequate or undefined pixel areas are avoided.

3.1.5. Image augmentation using Cycle-GAN deep learning model

The generative adversarial network (GAN) can learn the characteristics of incoming data and create similar data. There are two types of models: generative and discriminative models. The generative model aims to generate samples that get closer and closer to the true samples (G). By capturing the distribution of the genuine sample data, the generator can produce a sample that is comparable to the original training data with a noise z that follows a predefined distribution such as uniform or Gaussian distribution. The discriminator (D) is a binary classifier that determines how likely a sample is drawn from training data rather than generated data. The discriminator will output a high probability if the sample belongs to the true training data, and a low probability if it does not. The generator and discriminator alternately enhance their networks during training until Nash equilibrium is obtained. The generator can now generate samples with the same distribution as the original data, and the discriminator can detect the created samples with a 50% accuracy. The resulting generative model is used to create required new data. Cycle-GAN can learn the features of diseased turmeric leaf plants and healthy turmeric plants by training process and generating diseased spots on the surface of healthy turmeric leaves. This makes us generate new images even though have less amount of diseased turmeric leaf plants.

3.2. Pre-processing using median filter

After data augmentation input data needs to preprocess to remove the noise. This work applies median filter for noise removal**.** Median filtering is a nonlinear operation [14, 15]. This work utilized this as a part of picture handling to decrease salt and pepper and spot noise. The process of separating keep the sharpness of a picture edge while evacuating clamors. In this filter the neighboring pixels are ranked according to brightness and middle esteem turn into the new incentive for the focal pixel. The length of the filtering window is describe as n where signal length is N. The output of the filter is given by the function:

$$med(a_i) = \left\{ a_{k+1} n = 2k+1(odd) \quad \frac{[a_k + a_{k+1}]}{2} n = 2k(even) \right\} \quad (3)$$

Here $a_k$is the k-th maximum observed data and a_1, a_2, a_3...a_k are the observed data. It is the highlighting characteristic of the median filter that it eliminates the pulse noise, and local details remain intact. After this technique, the resulting image is then provided to the feature extraction block, where the GLCM is applied to the images to extract features.

3.3. Feature extraction using Gray Level Co-Occurrence Matrix (GLCM)

Once the images are get preprocessed it send to the feature extraction. GLCM is used in this work for feature extraction. The Grey-Level Co-occurrence Matrix (GLCM)) is helpful in estimating the image characteristics associated with second-order statistics. The texture features are computed as below [16]: Energy: Energy indicates the uniformity observed in the mammographic image. Generally, energy is computed from the value of the mean squared signal [17]. It is computed as below

$$Energy = \sum_{i,j=0}^{n-1} p(i,j)2 \quad (4)$$

Contrast: The contrast provides the measure of the difference between the least and the highest values of a set of pixels present in vicinity. It computes the amount of the local differences that exist in the image.

$$\text{Contrast} = \sum_{i,j=0}^{n-1} (i-j)^2 \text{p(i,j)} \quad (5)$$

Correlation: The correlation provides a measure of the correlation of a pixel with its neighbour over the entire image[18].

$$\text{Correlation} = \sum_{i,j=0}^{n-I} \frac{(i \times j)\, p(i,j) - u_i u_j}{\sigma_i \sigma_j} \quad (6)$$

σ2 = the difference of the intensities of all the reference pixels in the associations, which made their contribution to the GLCM, computed as follows:

$$\theta^2 = \sum_{i,j=0}^{N-1} p_{i,j}\text{(i-u)} \quad (7)$$

Homogeneity, Angular Second Moment (ASM): ASM utilized for measuring the homogeneity of the image

$$\text{Homogeneit} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \{p(i,j)\}^2 \quad (8)$$

Entropy: Entropy specifies the measure of the irregularity or complexity existing in the image. Entropy achieves the highest value if the values of P (i, j) are assigned quite evenly throughout the entire matrix. Entropy has ahigh but inverse correlation with Energy[19].

$$\text{Entropy} = -\sum_{i,j=0}^{n-1} p(i,j) log\ p(i,j) \quad (9)$$

Where ‘i’ specifies the rows of the GLCM matrix, ‘j’ refers to the columns of the GLCM matrix, ‘n’ specifies the number of gray levels and P(i, j) refers to the cell represented by the row and the column of the GLCM matrix. Depending on these assessments, the extractions of the texture features are obtained. .

3.4. Classification using Improved Yolov4

After feature extraction need the extracted features are classified to detect the diseases. In this work improved Yolov4 is used for classification. YOLO-v4 is a high-precision and real-time One-Stage object detection algorithm based on regression proposed in 2020, which integrated the characteristics

of YOLO-v1, YOLO-v2, YOLO-v3, etc., and achieved the current optimum in terms of detection speed and trade-off of detection accuracy. The model structure is shown in Figure 2, which consists of three parts: Backbone, Neck, and Prediction[20]. Combined with the characteristics of the ResNet structure, YOLO-v3 integrated the residual module into itself and then obtained Darknet53. Based on this, taking the superior learning ability of Cross-Stage Partial Network (CSPNet) into account, YOLO-v4 constructed the CSPDarkNet53.

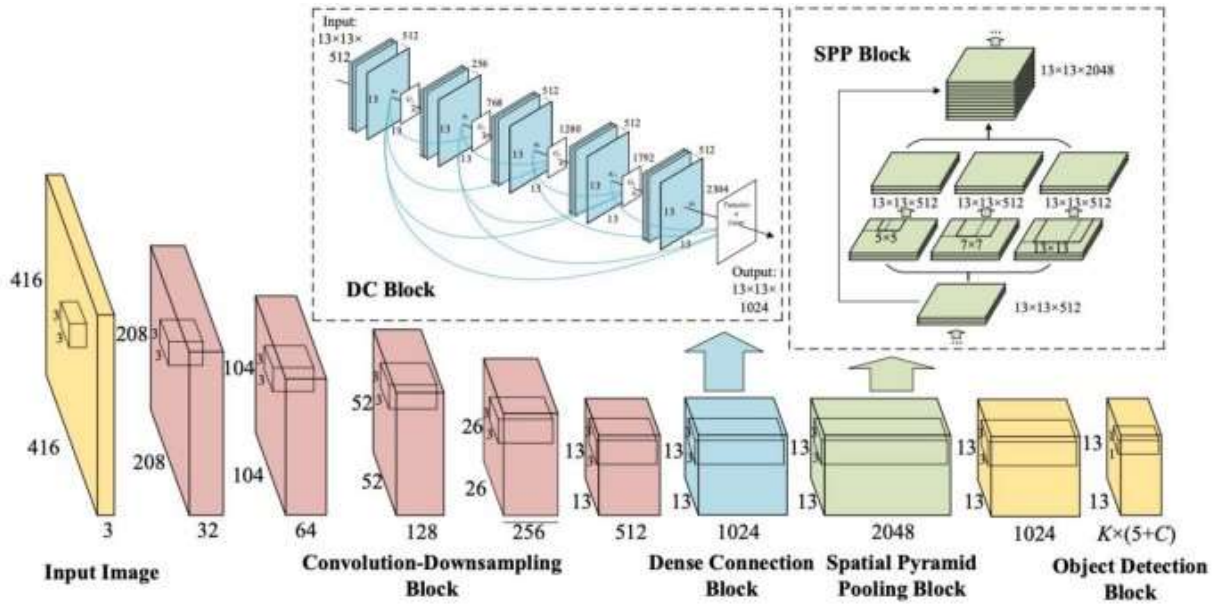

Figure 2.Yolov4 architecture

In the residual module, the feature layer is input and the higher-level feature information is output. This means the learning goal of the model in the ResNet module becomes the difference between the output and the input, thus realizing residual learning while reducing the parameters of the model and strengthening feature learning [21]. The Neck can be composed of the SPPNet and PANet. In SPPNet, firstly, the feature layer is convolved three times, and then the input feature layer is maximally pooled by using the maximum pooling cores of different sizes. The pooled results are concatenated firstly and then convolved three times, thus improving the network receptive field. PANet convolves the feature layers after the operation of Backbone and SPPNet and then up-samples them, that is, making the original feature layers double in height and width, and then concatenates the feature layers after convolution and up-sampling with the feature layers obtained by CSPDarkNet53 to realize feature fusion, and then down-sampling, compressing the height and width, and finally stacking with the previous feature layers to realize more feature fusion (five times). The Prediction module can make predictions by using the feature extracted from the network. Taking a 13×13 grid, for example, it is equivalent to divide the input picture into 13×13 grids, and then each grid will be preset with three prior frames[22]. The prediction results of the network will adjust the positions of the three prior frames, and finally, it will be filtered by the non-maximum suppression (NMS) algorithm to obtain the final prediction frame. YOLO-v4 proposed a new mosaic data augmentation method to expand the data set and introduced CIOU as the positioning loss function, which made the network more inclined to optimize in the direction of increasing overlapping areas, thus effectively improving the accuracy.In the actual complex environment, due to the external interference such as occlusion is still some shortcomings in the turmeric plant diseases detection directly using YOLO-v4. The main performances are as follows: There are still problems such as in the reasoning stage, the model adds gray bars at both ends of the image to prevent the image from

distorting, but too many gray bars increase the redundant information of the model. At the same time, the model has problems such as long training time, high calculation cost and overfull parameters. To solve these problems, this paper optimizes and improves the model based on YOLO-v4.

IMPROVED YOLO-V4 NETWORK

With the increasing number of layers of the convolutional neural network, the depth of the network is deepening, and the deeper network structure is beneficial for the extraction of object features. Thereupon, the semantic information of small objects is increased. The main improvements presented in this paper based on YOLO-v4 are as follows: The CSPDarkNet53 is improved into CSP1_X and CSP2_X, and so reduced network modules to reduce the parameters of feature extraction in the network model; using the CSP2_X module in Neck can increase information fusion.

Backbone Feature Extraction Network

The residual module introduced into YOLO-v4 is to enhance the learning ability of the network and reduce the number of parameters. The operation process of the residual module (Res-unit) can be summed up as follows. Firstly, perform 1 × 1 convolution; then 3 × 3 convolution; and weighting the two outputs of the module at last.

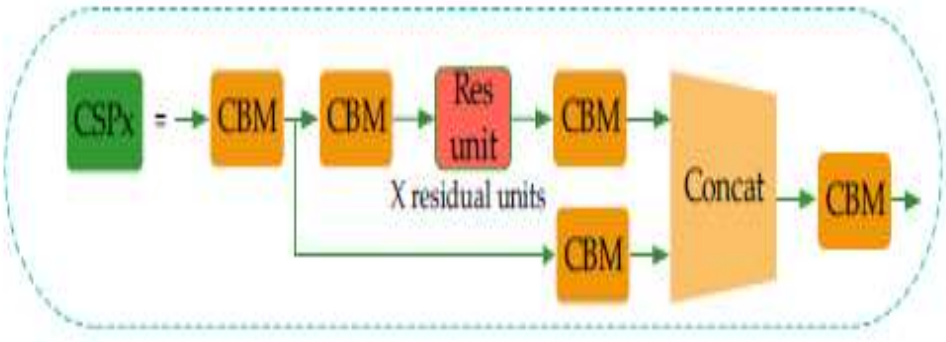

Figure 3. CSPDarkNet53 module structure

The purpose of weighting is to increase the information of the feature layer without changing its dimension information. In CSPDarkNet53, the set of feature layers of the image is input, and then convolution down-sampling is performed continuously to gain higher semantic information. Therefore, the last three layers of Backbone have the highest semantic information, and then the last three layers of features are selected as the input of SPPNet and PANet. The network structure of CSPDarkNet53 is shown in Figure 3. Although YOLO-v4 uses the residual network to reduce the computing power requirement of the model, its memory requirement still needs to be improved. Therefore, in this work, the network structure of CSPDarkNet53 of YOLO-v4 is improved to the CSP1_X module . Compared with CSPDarkNet53 in Figure 2, the improved network uses the H-swish activation function, as shown in Equation (10)

$$H - swish(X) = x\frac{ReLU(x+3)}{6} \quad (10)$$

As the Swish function contains the sigmoid function, the calculation cost of the Swish function is higher than the Re*LU* function, but the Swish function is more effective than the Re*LU* one. Howard used the H-swish function on mobile devices to reduce the number of accesses to memory by the model, which further reduced the time cost. Therefore, in this work, the advantages of the H-swish function are used to reduce the running time requirements of the model on condition of ensuring no gradient explosion, disappearance and other problems. At the same time, the detection accuracy of the model is advanced. In CSP1_X, the input feature layer of the residual block is divided into two branches. One is used as the residual edge for convolution operation. The other plays the role of the

trunk part, performs 1×1 convolution operation at first, then performs 1 × 1 convolution to adjust the channel after entering the residual block, and then performs the 3 ×3 convolution operation to enhance the feature extraction. At last, the two branches are concatenated, thus merging the channels to obtain more feature layer information. In this paper, three CSP1_X modules are used in the improved Backbone, where X represents the number of residual weighting operations in the residual structure. Finally, after stacking, a 1 ×1 convolution is used to integrate the channels. Experiments show that using this residual structure can make the network structure easier to optimize.

NECK NETWORK

The convolution neural network requires the input image to have a fixed size. In the past convolution neural network, the fixed input was obtained by cutting and warping operations, but these methods easily bring about problems such as object missing or deformation. To eliminate such problems, researchers proposed SPPNet to remove the requirement of fixed input size. To gain multi-scale local features, YOLO-v4 introduced the SPPNet structure based on YOLO-v3. In order to further fuse the multi-scale local feature information with the global feature information, add the CSP2_X module to the PANet structure of YOLO-v4 to enhance the feature extraction, which helps to speed up the flow of feature information and enhance the accuracy of the model. CSP2_X is shown in Figure 5. The common convolution operation is adopted in the Neck network in YOLO-v4, while the CSPNet has the advantages of superior learning ability, reduced computing bottleneck and memory cost. Adding the improved CSPNet network module based on YOLO-v4 can further enhance the ability of network feature fusion. This combined operation can realize the top-down transmission of deeper semantic features in PANet, and at the same time fuse the bottom-up deep positioning features from the SPPNet network, thus realizing feature fusion between different backbone layers and different detection layers in the Neck network and providing more useful features for the Prediction network.

IMPROVED NETWORK MODEL STRUCTURE

The improved network model is shown in Figure 8, in which three CSP1_X modules are used in the Backbone of the backbone feature extraction network, and each CSP1_X module has X residual units. In this work, considering the calculation cost, the residual modules are connected in series into the combination of X residual units. This operation can replace the two 3 × 3 convolution operations with a combination of 1×1 + 3×3 + 1×1 convolution module. The first 1 × 1 convolution layer can compress the number of channels to half of the original one and reduce the number of parameters at the same time. The 3 × 3 convolution layer can enhance feature extraction and restore the number of channels. The last 1 × 1 convolution operation restores the output of the 3 × 3 convolution layer, so the alternate convolution operation is helpful for feature extraction, ensures accuracy and reduces the amount of computation. The Neck network is mainly composed of the SPPNet and improved PANet. In this work, the SPPNet module enlarges the acceptance range of backbone features effectively, and thus significantly separates the most important contextual features. The high computational cost of model reasoning is mainly caused by the repeated appearances of gradient information in the process of network optimization. Therefore, from the point of view of network model design, this paper introduces the CSP2_X module into PANet to divide the basic feature layer from Backbone into two parts and then reduces the use of repeated gradient information through cross-stage operation. In the same way, the CSP2_X module uses the combination of 1 ×1 + 3 × 3 + 1 × 1 convolution module to reduce the computation cost and ensure accuracy.

The Prediction module uses the features extracted from the model to predict. In this work, the Prediction network is divided into three effective feature layers: 13 × 13 × 24, 26 × 26 ×24 and 52 ×52 ×24, which correspond to big object, medium object and small object, respectively. Here, 24 can be understood as the product of 3 and 8, and 8 can be divided into the sum of 4, 1 and 3, where 4

represents the four position parameters of the prediction box, 1 is used to judge whether the prior box contains objects, and 3 represents that there are three categories of diseases detection tasks.

4. RESULTS AND DISCUSSION

This section discusses the experimental results of the proposed IYOLOV4.Proposed model is implemented using mat lab and compared with existing OPNN and IYOLOV3-tiny models interns of precision, accuracy, recall and f-measure for the turmeric leaf database

Performance Metrics

1) Precision

Precision refers to the percentage of the results which are relevant and defined as

$$\text{Precision} = \frac{Truepositive}{truepositive + falsepositive} \quad (11)$$

2) Recall

Recall refers to the percentage of total relevant results correctly classified by the proposed algorithm which is defined as

$$\text{Recall} = \frac{Truepositive}{truepositive + FalseNegative} \quad (12)$$

3) Accuracy

Accuracy is the fraction of predictions this model got right. Formally, accuracy has the following definition

$$\text{Accuracy} = \frac{Truepositive + TrueNegative}{Total} \quad (13)$$

4) F measure

An F-score is the harmonic mean of a system's precision and recall values

$$2 \times [(\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})] \quad (14)$$

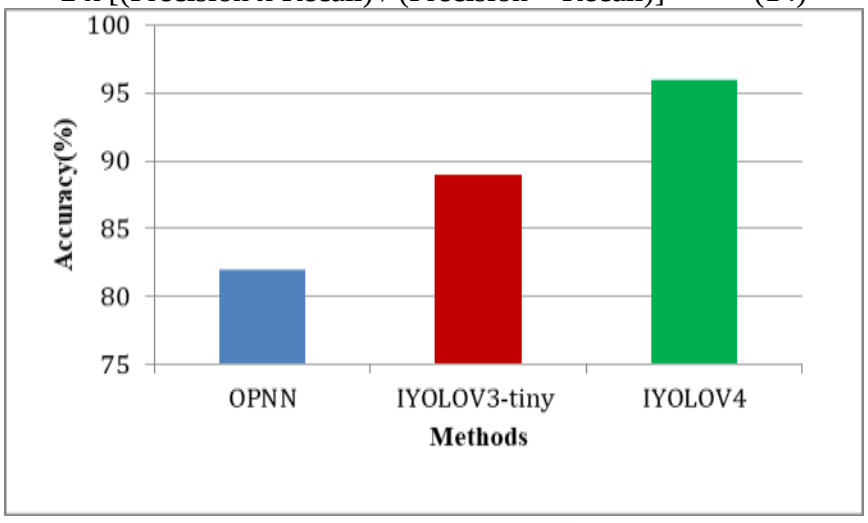

Figure 4.Accuracy results

Figure 4.Shows the accuracy performance metric comparison between existing OPNN and IYOLOV3-tiny methods and proposed IYOLOV4 for turmeric plant diseases classification. In the above figure X-axis represents the methods and the y-axis represents the accuracy results. This work using data augmentation by which accuracy increases. From the results it is concluded that the proposed IYOLOV4 model produces the higher accuracy results of 96% while the existing OPNN and IYOLOV3-tiny models produces only 82% and 89% accordingly.

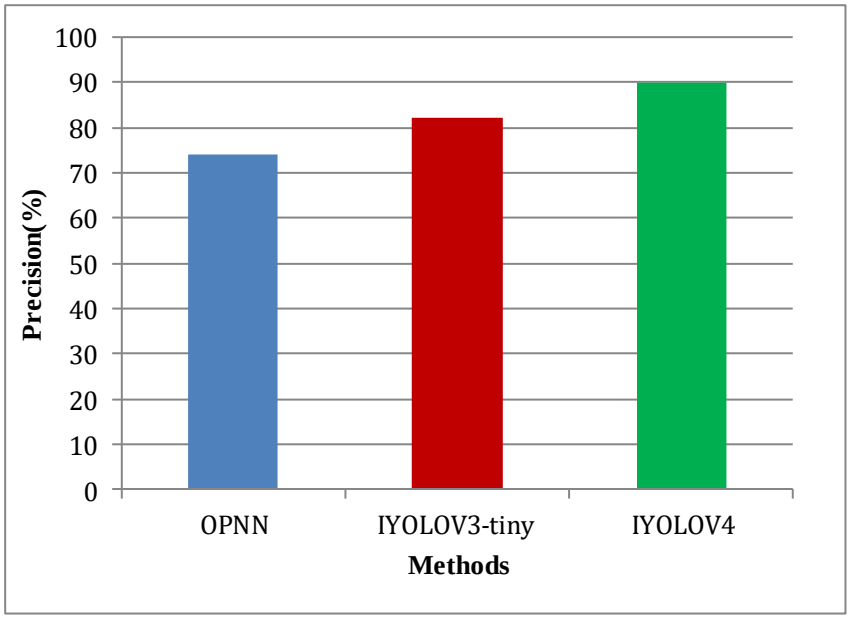

Figure 5 Precision results

Precision performance metric comparison between existing OPNN and IYOLOV3-tiny methods and proposed IYOLOV4 for turmeric plant diseases detection are shown in figure.5. In the above figure X-axis represents the methods and the y-axis represents the precision results. Proposed work using median filtering for noise removal and it increases the precision results. From the results it is concluded that the proposed IYOLOV4 model produces the higher precision results of 90% while the existing OPNN and IYOLOV3-tiny models produces only 74% and 82% accordingly.

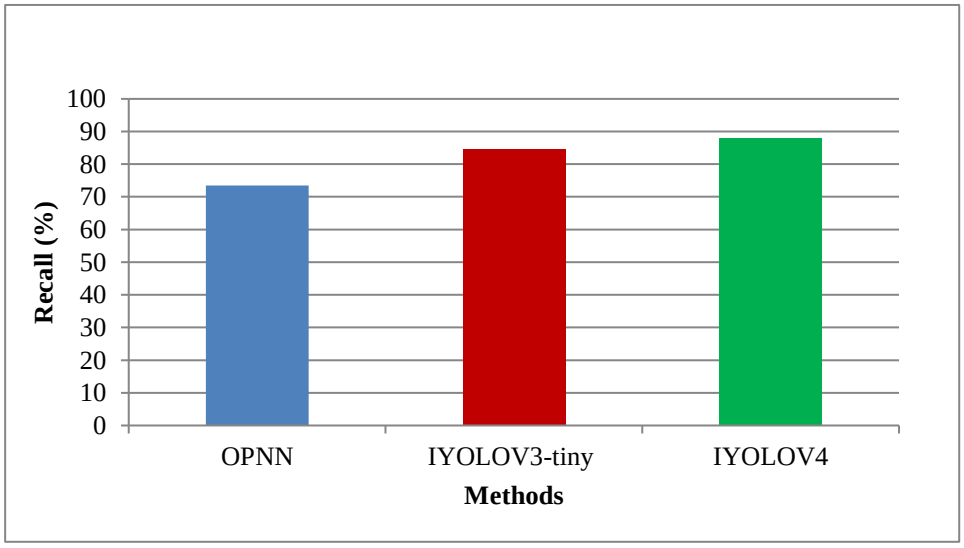

Figure 6 Recall results

Figure 6.Shows the performance comparison results for the existing OPNN and IYOLOV3-tiny methods and proposed IYOLOV4 for turmeric plant diseases detection interms of recall. In the above figure X-axis represents the methods and the y-axis represents the recall results. From the results it is concluded that the proposed IYOLOV4 model produces the higher recall results of 88% while the existing OPNN and IYOLOV3-tiny models produces only 73.5% and 84.5% accordingly.

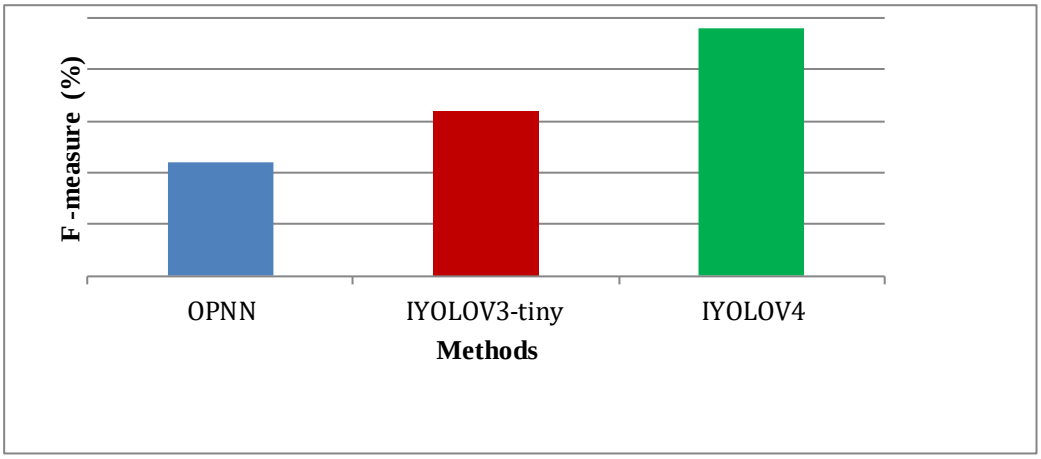

Figure .7 F -measure results

F-measure performance metric comparison between existing OPNN and IYOLOV3-tiny methods and proposed IYOLOV4 for turmeric plant diseases detection are shown in figure.7. In the above figure .X-axis represents the methods and the y-axis represents the f -measure results. From the results it is concluded that the proposed IYOLOV4 model produces the higher f -measure results of 89% while the existing OPNN and IYOLOV3-tiny models produces only 76% and 81% accordingly.

5. CONCLUSION AND FUTURE WORK

Disease identification plays a vital role in agricultural sector. Turmeric being a rhizomatous crop and well known for its therapeutic effects, monitoring such crops is crucial. The turmeric leaves are mainly exposed to diseases like Leaf Spot and Leaf Blotch. This work aimed to provide an improved model for automatic detection of turmeric plant diseases. In this work initially data augmentation is performed to improve the classifier accuracy using methods such as image rotation, image colour, image brightness transformation, motion blur transformation and Cycle-GAN deep learning model. Preprocessing is computed using median filtering based noise removal. After that texture features are extracted using Gray Level Co-Occurrence Matrix (GLCM) method. Major diseases in turmeric plants are detected using Improved YOLOV4 (IYOLOV4) model. Results show the effectiveness of the proposed model in terms of precision, recall, accuracy and f-measure. Results show that the proposed model achieves 96% accuracy which is better than other existing models. However proposed model still consumes more time for computation and this could be focused in future

REFERENCES

1. Vishnoi, V.K., Kumar, K. and Kumar, B., 2021. Plant disease detection using computational intelligence and image processing. Journal of Plant Diseases and Protection, 128, pp.19-53.
2. Velmurugan, S., Reddy, K.R., Rahul, S.G., Vardhan, S., Subitha, D. and Vignesh, S.K., 2022, March. A Comparative Computational Analysis of VGG16and VGG19 in Prediction of Turmeric plant Disease. In 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS) (Vol. 1, pp. 130-134). IEEE.
3. Kurale, N.G. and Vaidya, M.V., 2018, July. Classification of leaf disease using texture feature and neural network classifier. In 2018 International Conference on Inventive Research in Computing Applications (ICIRCA) (pp. 1-6). IEEE.
4. Golhani, K., Balasundram, S.K., Vadamalai, G. and Pradhan, B., 2018. A review of neural networks in plant disease detection using hyperspectral data. Information Processing in Agriculture, 5(3), pp.354-371.
5. Barbedo, J.G.A., 2016. A review on the main challenges in automatic plant disease identification based on visible range images. Biosystems engineering, 144, pp.52-60.

6. Singh, V. and Misra, A.K., 2017. Detection of plant leaf diseases using image segmentation and soft computing techniques. Information processing in Agriculture, 4(1), pp.41-49.
7. Suryawanshi, A.S. and Khurjekar, M.J., 2021, September. Aerial Imagery for Plant Disease Detection by Using Machine Learning of Typical Crops in Marathwada. In 2021 International Conference on Computing, Communication and Green Engineering (CCGE) (pp. 1-5). IEEE.
8. Kuricheti, G. and Supriya, P., 2019, April. Computer vision based turmeric leaf disease detection and classification: a step to smart agriculture. In 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI) (pp. 545-549). IEEE.
9. Albattah, W., Nawaz, M., Javed, A., Masood, M. and Albahli, S., 2022. A novel deep learning method for detection and classification of plant diseases. Complex & Intelligent Systems, pp.1-18.
10. Tulshan, A.S. and Raul, N., 2019, July. Plant leaf disease detection using machine learning. In 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-6). IEEE
11. Ramesh, S., Hebbar, R., Niveditha, M., Pooja, R., Shashank, N. and Vinod, P.V., 2018, April. Plant disease detection using machine learning. In 2018 International conference on design innovations for 3Cs compute communicate control (ICDI3C) (pp. 41-45). IEEE.
12. Akanksha, E., Sharma, N. and Gulati, K., 2021, January. OPNN: optimized probabilistic neural network based automatic detection of maize plant disease detection. In 2021 6th international conference on inventive computation technologies (ICICT) (pp. 1322-1328). IEEE.
13. Tm, P., Pranathi, A., SaiAshritha, K., Chittaragi, N.B. and Koolagudi, S.G., 2018, August. Tomato leaf disease detection using convolutional neural networks. In 2018 eleventh international conference on contemporary computing (IC3) (pp. 1-5). IEEE.
14. Erkan, U., Gökrem, L. and Enginoğlu, S., 2018. Different applied median filter in salt and pepper noise. Computers & Electrical Engineering, 70, pp.789-798.
15. Wang, Y., Wang, J., Song, X. and Han, L., 2016. An efficient adaptive fuzzy switching weighted mean filter for salt-and-pepper noise removal. IEEE signal processing letters, 23(11), pp.1582-1586.
16. Abbas, Z., Rehman, M.U., Najam, S. and Rizvi, S.D., 2019, February. An efficient gray-level co-occurrence matrix (GLCM) based approach towards classification of skin lesion. In 2019 amity international conference on artificial intelligence (AICAI) (pp. 317-320). IEEE.
17. Hlaing, K.N.N. and Gopalakrishnan, A.K., 2016, January. Myanmar paper currency recognition using GLCM and k-NN. In 2016 Second Asian Conference on Defence Technology (ACDT) (pp. 67-72). IEEE.
18. Reddy, A.M., SubbaReddy, K. and Krishna, V.V., 2015, October. Classification of child and adulthood using GLCM based on diagonal LBP. In 2015 International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT) (pp. 857-861). IEEE.
19. Kumar, C., Chauhan, S. and Alla, R.N., 2015, April. Classifications of citrus fruit using image processing-GLCM parameters. In 2015 International Conference on Communications and Signal Processing (ICCSP) (pp. 1743-1747). IEEE.
20. Yijing, W., Yi, Y., Xue-fen, W., Jian, C. and Xinyun, L., 2021, May. Fig fruit recognition method based on YOLO v4 deep learning. In 2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON) (pp. 303-306). IEEE.
21. Wang, L., Zhou, K., Chu, A., Wang, G. and Wang, L., 2021. An improved light-weight traffic sign recognition algorithm based on YOLOv4-tiny. IEEE Access, 9, pp.124963-124971.