Received 29 September 2022, accepted 24 December 2022, date of publication 27 December 2022, date of current version 30 December 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3232751

RESEARCH ARTICLE

Handling Severity Levels of Multiple Co-Occurring Cotton Plant Diseases Using Improved YOLOX Model

SEROSH KARIM NOON[1,2], (Graduate Student Member, IEEE), MUHAMMAD AMJAD[1], MUHAMMAD ALI QURESHI[1], (Senior Member, IEEE), AND ABDUL MANNAN[3]

[1]Department of Electrical Engineering, The Islamia University of Bahawalpur, Bahawalpur 63100, Pakistan
[2]Department of Electrical Engineering, NFC Institute of Engineering and Technology, Multan 59060, Pakistan
[3]Department of Biomedical Engineering, NFC Institute of Engineering and Technology, Multan 59060, Pakistan

Corresponding author: Serosh Karim Noon (seroshkarim@nfciet.edu.pk)

ABSTRACT Automatic detection of plant diseases has emerged as a challenging field in the last decade. Computer vision-based advancements have helped in the timely and accurate identification of diseases, making possible an appropriate treatment and hence ensuring an increased yield. Diseases attack in different formations on a plant; the most severe being multiple diseases appearing on a single leaf. Moreover, as various diseases progress, they generate similar-looking symptoms making the task of identification further difficult. This work addresses these two problems with the help of an improved YOLOX model. We propose a modified Spatial Pyramid Pooling (SPP) layer to effectively extract relevant features at various scales from the training data. It is achieved by concatenating multilevel features pooled from smaller to larger scales. To enhance the generalization capability of the design, various skip connections are also introduced. To improve the network convergence and detection accuracy, α_{IoU} based regression loss function was employed. A dataset composed of $1,112$ cotton plant images with co-occurring diseases along with their progressive severity levels was collected from the Southern Punjab region of Pakistan. Apart from healthy images, the dataset comprises three severity stages of cotton leaf curl with co-occurring cotton sooty mold stress on a single leaf image. Experimental results revealed that our proposed improved SPP-based YOLOX-s model achieved 73.13% mAP on our self-collected dataset and achieved 3.27% better test accuracy than the original YOLOX model.

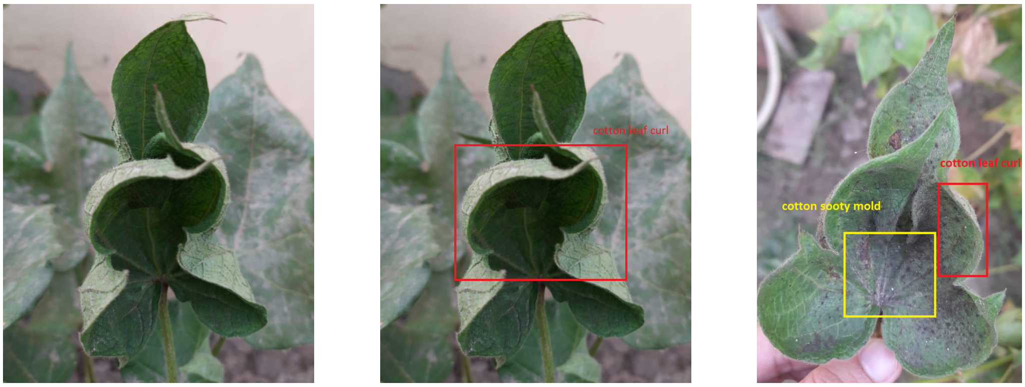
INDEX TERMS Plant disease, YOLOX model, disease severity classification, multiple stress, cotton plant.

I. INTRODUCTION

Cotton is an important cash crop in Pakistan grown on 15% of the total cultivated area in the country. Apart from its domestic consumption, the country's cotton production and export contribute a rich share of the world's cloth economy. During its lifespan, the crop is affected by many diseases. Sometimes, a leaf is affected by more than one disease whose symptoms are alike too. Disease severity symptoms and the presence of multiple stresses can be misjudged even by an expert pathologist.

With the advent of artificial intelligence and computer vision techniques, precision farming techniques have been revolutionized. Several machine learning and deep learning models have shown remarkable performance in plant disease detection systems [1]. Combined with transfer learning, several researchers have used deep learning-based feature extraction and classification tasks on self-collected or publicly available datasets. Various studies have aimed to achieve near-ideal performance in detecting disease symptoms automatically and suggest the use of pesticides or preventive measures. For the last few years, well-known deep learning architectures like region-based convolutional networks [2], single shot detectors [3], and region proposal networks [4] have been used in the field plant leaf disease detection with necessary modifications [3]. A sample output of a meta-deep learning architecture detecting a plant leaf disease is shown in Figure 1.

The associate editor coordinating the review of this manuscript and approving it for publication was Gangyi Jiang.



(a) Classification of disease in image

(b) Classification and localization of the disease part

(c) Detection with localization of diseased parts

FIGURE 1. Classification, localization, and detection tasks described with the help of sample images.

Almost all previous studies have either used the famous PlantVillage dataset [3] or their self-collected datasets [5], [6].

However, very few studies have discussed (1) stages of disease progression and (2) the presence of more than one biotic and abiotic agent attacking a plant leaf simultaneously. These situations pose a challenging situation for both manual and automatic detection strategies to identify the type of infection and the exact localization of disease symptoms. As an example, various stages of cotton leaf curl disease found on the cotton plant are shown in Figure 1(a). Further, co-occurring symptoms of cotton leaf curl and cotton sooty mold are shown in Figure 1(b).

In order to detect stages of diseases and segregate diseases with overlapping and co-occurring symptoms, we, in this study, present a YOLOX-s based detection framework employing a modified Spatial Pyramid Block to capture fine spatial features after aggregating them with local features. To address the problem with better accuracy we improved the spatial pyramid pooling block by concatenating feature maps at low-level scales. The integrity of features was enhanced by adding the original size input feature vector. The detection performance was further optimized by the using α-IoU regression loss function.

The main contributions of this work are as follows:

- To detect more than one disease on a single leaf plant, an improved YOLOX-s model based on a modified Pyramid pooling module (SPP) layer is presented. It collects spatial details at local multi-scale levels to capture the desired information more effectively.
- For better generalization and convergence, in situations where multiple diseases appear on a single plant leaf, we used α-IoU loss as bounding box regression for multiple disease localization.
- A dataset of self-collected real field images along with their segmented versions is presented. The images contain the progression of diseases and the presence of multiple diseases per leaf.

II. RELATED WORK

Over the last decade, various traditional machine learning [7] and deep learning techniques have been used to develop automatic recognition of plant diseases [8]. Many effective feature extraction techniques have been proposed. [9] for efficient disease diagnosis based on color, shape, and texture [3].

The use of computer vision-based techniques in the area of plant leaf diseases can be divided into (1) detection and (2) recognition. The former refers to the task of locating the diseased part of the leaf in a scene containing the healthy part and background too. The latter, however, is referred to when the type of disease is to be decided given only the segmented portion containing stress(es). Research literature published in the last decade is full of both types of approaches applied in the field.

Table 1 provides highlights of some recent and state-of-the-art approaches in the fields of plant leaf disease detection and recognition. These approaches will be discussed in this section in detail with respect to: (1) the type of application addressed, (2) the technique used, (3) the contribution, (4) the dataset used, and (5) the limitation of the work.

In the last few years, the use of deep learning became prevalent in this area of research. Employing transfer learning and data augmentation has facilitated the implementation of deep learning models on a variety of hardware including CPUs and GPUs. The work presented by Abayomi et al. [10] used color space data augmentation techniques to train the MobileNetV2 model. The authors trained the model for low and high-quality cassava leaf disease images to compare the classification performance. It was reported that the classification accuracy is reduced for low-quality images. Authors [11] have also used high-quality images to locate the diseased area based on color difference. Advanced machine learning classifiers like bagged tree ensemble achieved an overall accuracy of 99% to distinguish diseased areas based on color and textural features. Since Computer vision-based plant disease detection systems are supposed to locate and identify the diseased part of a plant automatically, the use of versatile deep-learning meta-architectures is found in the literature.

Region-based Convolutional Neural networks (RCNNs) were the first deep learning algorithms used for object detection in general [3]. Just like Fast RCNN [12], Faster-RCNNs, R-FCNNs [3] generate a series of candidate frames for targeted objects and then classify each frame separately. Zhou et al. [13] proposed a rice disease detection model using the Faster R-CNN model fused with the K-means clustering algorithm (FCM-KM). OTSU algorithm and multilevel median filters were used to perform segmentation and noise removal tasks respectively. The algorithm achieves an inference speed of 0.52 s by using a two-stage detector. A similar two-stage detector Mask RCNN was used by Rehman et al. [14] to detect diseased areas after contrast stretching. Feature extraction of enhanced regions is performed by CNN which was later classified. Better accuracy was obtained after the selection of the best features using Kapur's entropy. In another notable work, authors used faster R-CNN with an improved backbone [15] to detect various rice diseases in still video. The improved two-stage model shows better results in terms of recall and accuracy but with long detection times when compared with other models like YOLOV3. The model used in [16] achieved significant improvement in mAP using an enhanced anchor box approach based on a faster RCNN model for weed detection.

Overall optimized performance with better inference time can be obtained without generating region proposal network as in single shot detection (SSD). They employ predefined boxes to determine the probability of an object in an image. Such an improved model with the Inception module was used to extract feature information and Rainbow concatenation to enhance the detection of apple diseased spots [17]. The model incorporated VGG with the inception module to classify disease with a mAP of 78.8%. On the other hand, we have YOLO (You Only Look Once) [18] models that are also single-stage detectors that only require a single forward propagation to locate and classify objects in a scene. It detects by first dividing an image in a grid and then predicts the probability of an object and its class for each bounding box. A similar YOLO framework was adopted by Wang et al. [19] who integrated the DenseNet block for feature extraction in the YOLO framework to detect tomato disease in complex background conditions. Further details are mentioned in the related work comparison Table 1. Bacterial leaf spots were successfully detected using the YOLOV5 model. The results were also compared with YOLOV3, YOLOV4, SSD, and other two-stage detectors [20].

In the YOLO series, a recently introduced modification namely YOLOX [26] has achieved a significant improvement in the area of object detection. YOLOX uses the baseline framework of YOLOV3-Darknet with SPP Layer to enhance feature extraction. YOLOX is used to strip leaves from the complex background and semantic segmentation is performed using DeepLabV3+ and UNet to identify kiwi leaf diseases [21]. An improved YOLOv3 algorithm is proposed by integrating Cross Stage Partial Network (CSPNet) and Sigmoid Linear Unit (SiLU) Activation to detect colonic polyp [27]. The CIOU loss function [28] is also used to make the model robust for real-time polyp detection. Accurate recognition of plant disease was performed by Chen et al. [22] using the YOLOv5 model. To improve the accuracy and the number of parameters, the SE module and InvolutionBottleneck were used. Further, diseased lesions on paddy leaves were segmented using a hybrid deep-learning model with ResNet architecture and a YOLO classifier. The model parameters were further optimized by using the Fitness Sorted-Shark Smell Optimization technique [23].

A notable work addressed fine-grained recognition of strawberry leaf disease severity classification in real-field conditions. Leaf location in complex conditions was marked by Faster RCNN and severity estimation was performed using Siamese network [24]. Detecting multiple lesion spots in a single frame and disease progression are challenges faced by the researchers. Due to overlapping disease spots and symptoms, multiple diseases are hard to be detected both manually and with the use of automatic image processing techniques [29]. The existence of multiple diseases on a single cucumber leaf was effectively addressed by [25] using the EfficientNet-B4 model with 96% classification accuracy. The authors used a Ranger optimizer to identify similar-looking symptoms.

Although the techniques found in the literature (mentioned in Table 1) have used single-stage detectors with decent recognition performance. Most of these studies have focused on segmenting lesions or the identification of spots. The tasks of accurately identifying varying severity stages of a single disease and multiple stresses on a single leaf are still the least addressed. Therefore, we propose in this research, an application-specific improved YOLOX model to detect overlapping and co-occurring plant leaf diseases.

III. THE PROPOSED TECHNIQUE

This section describes complete details of the proposed technique used to address the problems of (1) symptoms of disease progression and (2) the presence of multiple diseases on a single leaf. The input data is pre-processed as the first step to eliminate unnecessary background details and to ensure a balance between classes. Next, we describe the details of the proposed deep learning model designed to tackle especially the two important scenarios mentioned above.

A. PRE-PROCESSING

Since leaf images were captured in real field conditions, they contained background information including soil and tree branches, etc. Extracting only the leaf area containing one or more diseases and removing the background information is supposed to make the task of the proposed detection mechanism easier [30]. Several techniques are in place to remove unnecessary background information in real-world scenes [9]. GrabCut [31] is a simple and efficient machine learning-based technique capable of removing undesired background information with minimal manual settings. The background is eliminated based on graph cuts. Based on

TABLE 1. Some notable efforts in the field of plant leaf disease detection.

Ref.	Model	Dataset	Novelty/salient feature	Weighted Measure	Compared with
[10]	MobileNetV2	Cassava dataset (9430)	Data augmentation technique (Convolution of orthogonal functions with Probability density function)	99.7% Classification accuracy	baseline model trained on high quality images
[11]	Bag Tree Ensemble Classifier	Guava high quality dataset	Color and textual features applied on classifiers	99% Accuracy color features	KNN, Cubic SVM, Boosted tree
[13]	Fusion of k-means with Faster R-CNN	Rice dataset (3010)	Fused R-CNN model	Average accuracy 97.2%	Faster R-CNN
[14]	Mask RCNN, Kapur entropy, SVM	Plant Village	Use of hybrid contrast stretching scheme to detect infected regions	99.1% accuracy	SVM, LDA, KNN
[15]	Faster RCNN	Self collected Rice disease& Pests (5320 images) and video frames (4290)	Custom CNN as framework with local response normalization and max pooling layer to extract features	Video Precision 87.2 %, Recall 68.0% and F1 76.4 (for untrained rice videos)	VGG16, ResNet-50, ResNet-101 backbone system and YOLOv3
[16]	Faster RCNN (ResNet 101)	Deepweed dataset	Enhanced anchor box technique to identify weed	96.02%mAP	SSD, RFCN, YOLOv4
[17]	SSD	Apple leaf disease dataset (26,377)	VGG with inception module and Rainbow Concatenation in SSD	78.80% mAP23.13 FPS	Faster R-CNN, SSD, DSSD, RSSD
[19]	YOLOv3	Self collected tomato dataset(15000)	Addition of dense module and K-means algorithm to cluster the anchor boxes	mAP 96.41% detection time 20.28	SSD, Faster R-CNN and the original YOLOv3
[20]	YOLOv5	Bell pepper images PlantVillage	YOLOv5 model trained for PlantVillage images was tested on real-field images	Training 20: mins, &inference time: 20ms, Size of model 27 MB	YOLOv4
[21]	YOLOX, DeepLabv3, ResNet	Real field kiwifruit leaf images(2000)	YOLOX to strip leaves from background. DeepLabv3+ for lesion cutting, ResNet for classification of lesions	Test accuracy +1% than default deeplabv3	Original DeepLabv3
[22]	YOLOv5	Rubber tree database (2375)	SE module & Involution Bottleneck	70% mAP 5.4% higher than YOLOv5	YOLOv5, YOLOX nano
[23]	YOLO with ResNet	UCI & Mendely paddy disease dataset	K-means clustering for segmentation,Fitness Sorted-Shark Smell Optimization with YOLO classifier	97.67% accuracy & 98.79 F1-score	PSO-Res-Yolo,GWO-Res-Yolo
[24]	Faster RCNN & Siamese Network	self collected strawberry leaf dataset	Faster RCNN clips strawberry patches and Siamese network identify the patch	96.97%accuracy, 88.83% accuracy on new dataset	YOLO, SSD
[25]	EfficientNet	Cucumber dataset (2816)	Ranger optimizer with EfficientNet-B4 model to classify multiple stress on single leaf	Classification accuracy: 96%	AlexNet, GoogleNet, ResNet

a window provided by the user, everything outside the window is considered the definite background, whereas, inside the window, there can be both the foreground and the background. A Gaussian Mixture Model (GMM), given in Equation 1, is then used to estimate the distribution of colors pertaining to the foreground and the background, presuming them to be normally distributed.

$$p(x) = \sum_{i=1}^{k} p(C_i|x) = \sum_{i=1}^{k} p(x|C_i)p(C_i) \tag{1}$$

where k represents the number of Gaussian distributions for foreground and background each.

Another parameter γ is defined which is 1 for foreground and 0 for background pixels. For each pixel n in the foreground, we define:

$$k_n = arg_{min} D_n(\gamma_n, k_n, \theta, \mathbf{x}_n) \tag{2}$$

Now the parameters θ of the GMM are estimated as:

$$\theta = arg_{min} U(\gamma, k, \theta, x) \tag{3}$$

In the next step, a Markov random field (MRF) is constructed on the labels assigned by the GMM. The cost function of the MRF is computed based on the connected regions with the same label. Finally, a graph cut optimization is run to divide the vertices into two groups namely the foreground and the background:

$$min\, E(\gamma, k, \theta, z) \tag{4}$$

To fine-tune the background removal task, the process is repeated until convergence. The average processing time per image for background removal was noted to be 4.12 seconds on the hardware used for experimentation.

A screenshot of a sample image processed for background removal using the standard GrabCut algorithm is given in Figure 2. The algorithm was found successful to remove the background information outside the infected leaf.

The images after background removal are annotated and are later subject to image augmentation techniques like flip, rotate, and brightness enhancement. This helps in improving the training performance of the model as well as to combat overfitting as discussed in Section IV-A in detail.

B. THE PROPOSED YOLOX-BASED OBJECT DETECTION MODEL

YOLOX [26] is an anchor-free detector that has shown remarkable performance in terms of both speed and accuracy. The head of YOLOX is decoupled from the original detector to achieve a better convergence during training [32]. Its anchor-free implementation reduces the number of trainable parameters significantly. Moreover, the label assignment strategy is further improved using SimOTA [26] which helps in addressing the Optimal Transport (OT) problem [21] in addition to reducing the training time.

For the application addressed in this paper, we come up with an enhanced version of the YOLOX model that makes it suitable for detecting (1) multiple disease symptoms on a single leaf separately and (2) severity levels of disease during its progression. Further, the proposed YOLOX model has only 8.9M parameters and 26.77G FLOPS.

A block diagram of the proposed detection scheme is shown in Figure 3. It is an improved version of the basic YOLOX model. The main part of the proposed model is the backbone CSP darknet used for carrying out feature extraction. Enhance feature fusion is performed in the feature pyramid network (FPN) part. Features obtained from three feature layers are up-sampled to perform feature fusion and are later down-sampled to classify in the classification and regression tasks [27]. The focus module is one of the prime components to extract features effectively. In the process to retain object feature information, the images at the input are sliced into four parts and are concatenated to visualize the features in depth. The deep features are extracted more precisely in the next step namely the BottleneckCSP layer. The feature maps are passed through convolution, respective batch normalization, and activation operations. Advanced data augmentation techniques like Mosaic and mix-up are used during training to learn smaller and overlapping symptoms [33]. Multiple detections are avoided using the non-maximal suppression (NMS) technique.

The output of FPN is fed into YOLOX decoupled head at three different scales. The decoupled head as shown in Figure 3 holds the information of class, location, and objectness in three separate tensors. Since there are three outputs, we will deal with three different loss functions. Binary Cross Entropy (BCE) loss [34] with logits defines the modeling error between the ground truth and the predicted class. All positive predictions are sorted out using a sigmoid activation function [26]. The output of the regression branch is $H \times W \times 4$ which predicts the bounding box coordinates (x, y, w, h). YOLOX uses the IoU metric to predict bounding box outputs and compares it with the ground truth.

C. BOUNDING BOX REGRESSION

Localization and classification of objects are two vital steps in computer vision-based applications. The degree of correctness with which a machine learning model localizes an object in an image/scene is evaluated using the loss function [35]. Conventional single-stage and two-stage detectors use the bounding box regression technique for this purpose. Earlier detection models used l_n-norm losses; which have recently been replaced by Intersection over Union (IoU) localization losses computed using a simple formula given in Equation 5. Here, A is the ground truth box and B is the area of the predicted box. If the two boxes overlap completely, the IoU is 1.

$$IoU = \frac{A \cap B}{A \cup B} \tag{5}$$

However, IoU losses suffer from the vanishing gradient problem which in turn slows down the model's convergence. The situation occurs as the predicted boxes do not accurately

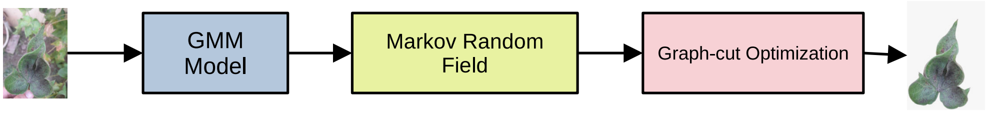


FIGURE 2. Block diagram describing the standard GrabCut algorithm. It was adopted in the proposed work to eliminate undesired background information.

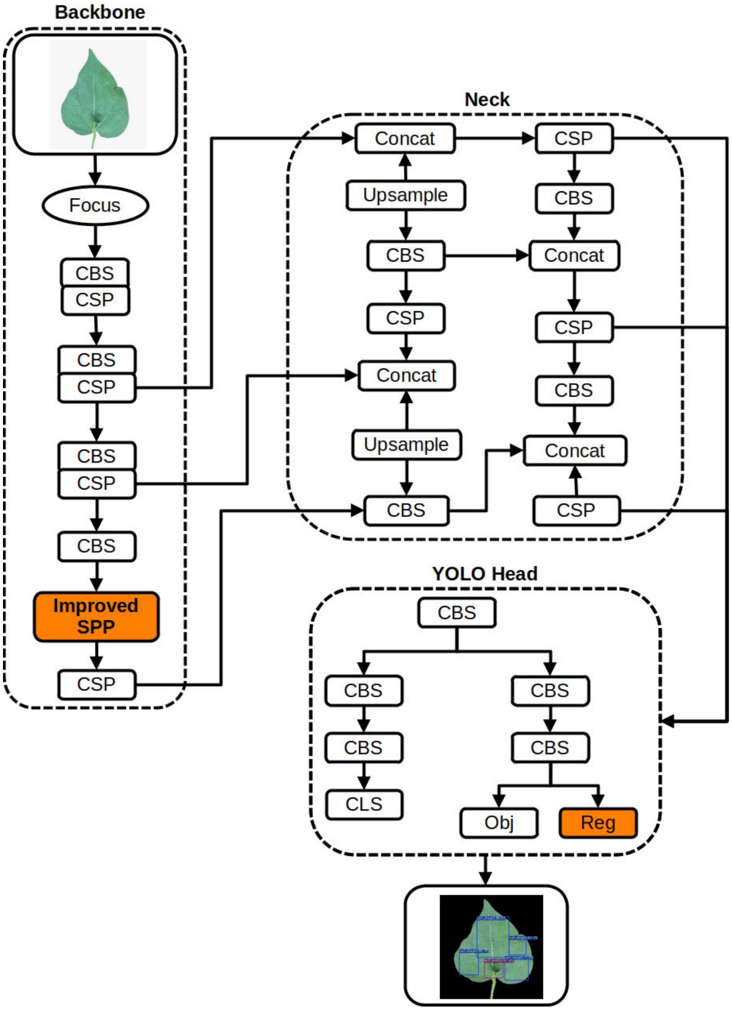


FIGURE 3. Architecture of the proposed YOLOX model with improved SPP block highlighted as yellow.

overlap the ground truth boxes and hence lead to inaccurate results. To track the objects more precisely, several improved bounding box losses were proposed based on different metrics. The GIoU (Generalized Intersection over Union), DIoU (Distance Intersection over Union), CIoU (Complete IoU), and EIoU (Efficient IoU). These loss functions use different metrics to reflect the degree of overlap between the target and the anchor boxes. For instance, DIoU and CIoU take into account the distance metric, whereas, GIoU not only focuses on the overlapping area but also takes into consideration the non-overlapping area.

In this work, we have employed a newly proposed family of loss functions known as α-IoU [36] due to its better convergence and detection accuracy characteristics. α-IoU loss is a modified version of the vanilla IoU loss, $\mathcal{L}_{IoU}$, defined as:

$$\mathcal{L}_{IoU} = 1 - IoU \tag{6}$$

By transforming and generalizing Equation 6, the α-IoU loss, $\mathcal{L}_{\alpha-IoU}$, is given as:

$$\mathcal{L}_{\alpha-IoU} = \frac{1 - IoU^{\alpha}}{\alpha}, \quad \alpha > 0 \tag{7}$$

where α is a modulating factor that can take only positive values. However, we can have the following two cases:

$$\mathcal{L}_{\alpha-IoU} = \begin{cases} -log(IoU), & \alpha \to 0, \\ 1 - IoU^{\alpha}, & \alpha \nrightarrow 0. \end{cases} \tag{8}$$

As we have used the case of $\alpha \geq 1$, the further generalized form is given as:

$$\mathcal{L}_{\alpha-IoU} = 1 - IoU^{\alpha_1} + P^{\alpha_2}(B, B^{gt}) \tag{9}$$

where $\alpha_1 > 0$ and $\alpha_2 > 0$ and P^{α_2} denotes the penalty term of predicted box B and ground truth box B^{gt}. The Equation 9 can be used to transform the IoU-based losses. We used the $\mathcal{L}_{CIoU}$ transformed to $\mathcal{L}_{\alpha-CIoU}$ using α and penalty parameter $P^{\alpha}(B, B^{gt})$ is calculated for predicted and ground truth bounding box as mentioned in Equation 10.

$$\mathcal{L}_{\alpha-CIoU} = 1 - IoU^{\alpha} + \frac{\rho^{2\alpha}(b, b^{gt})}{c^{2\alpha}} + (\beta v^{\alpha}) \tag{10}$$

where b, b^{gt} are the central points of the bounding boxes B, B^{gt} with ρ as the euclidean distance and c is the diagonal length of the smallest box and $v = \frac{4}{\pi^2}(arctan\frac{\omega^{gt}}{h^{gt}} - arctan\frac{\omega}{h})^2$ with $\beta = \frac{v}{(1-IoU)+v}$ where $\frac{\omega^{gt}}{h^{gt}}$ and $\frac{\omega}{h}$ are aspect ratios of ground truth box and predicted box respectively. When $\alpha = 1$ all the bounding box regression losses can be used in their original versions. $\mathcal{L}_{\alpha}$ when used with $\alpha > 1$ schemes adapt and reweigh its gradient and loss values accordingly. The reweighing property increases as the α value increases [36]. This adaptive and relative reweighing helps the model to converge better and achieve better performance when compared with all baselines IoUs. The potential of $\mathcal{L}_{\alpha-IoU}$ to attain better generalization and high average precision by applying a unified modulating factor made us apply it to our multiple disease localization models.

D. IMPROVED SPATIAL PYRAMID POOLING BLOCK

Figure 4 (a) shows the original structure of the Spatial Pyramid Pooling (SPP) block used to pool important feature information extracted in the backbone of the original YOLOX model. To extract more meaningful features in case of overlapping diseased regions, we propose in this work a modified SPP block for our Improved YOLOX model. The internal structure of our proposed Improved SPP block is given in Figure 4 (b). The input to the block is passed through a $Conv_1$ to perform channel-wise pooling by reducing the size of the input feature map to 512. The resultant $20 \times 20 \times 512$ tensor is fed to the next layer where features are extracted at multiple spatial scales using a number of max pooling operations performed using varying kernel sizes. The max pooling operation is mathematically described in Equation 11. Here, H and W are the height and width of the channel and H_k and W_k are the height and width of the respective kernel.

The proposed improved SPP block contains an additional parallel pooling block. The smaller pooling kernel pools deep features at a smaller receptive field.

$$O_{i,j} = max(H, W, H_k, W_k)H_k \in HW_k \in W. \tag{11}$$

The output tensor sizes of all four MaxPool layers are the same; calculations are given in Equation 12. Here, n_{in} and n_{out} are the sizes of the input and out feature maps respectively, k is the kernel size, p is the padding size and s is the stride.

TABLE 2. Parameters of various convolutional layers in the improved SPP block.

Layer	Kernel	Stride	Padding	Output Channels
$Conv_1$	1×1	1	0	512
$MaxPool^3$	3×3	1	1	512
$MaxPool^5$	5×5	1	2	512
$MaxPool^7$	7×7	1	3	512
$MaxPool^9$	9×9	1	4	512
$Concat$	-	-	-	2048
$Conv_2$	1×1	1	0	1024
Add	-	-	-	1024

As can be seen in Table 2, varying padding sizes were used to preserve the size of the output feature map.

$$n_{out} = \left\lfloor \frac{n_{in} + 2p - k}{s} \right\rfloor + 1 \tag{12}$$

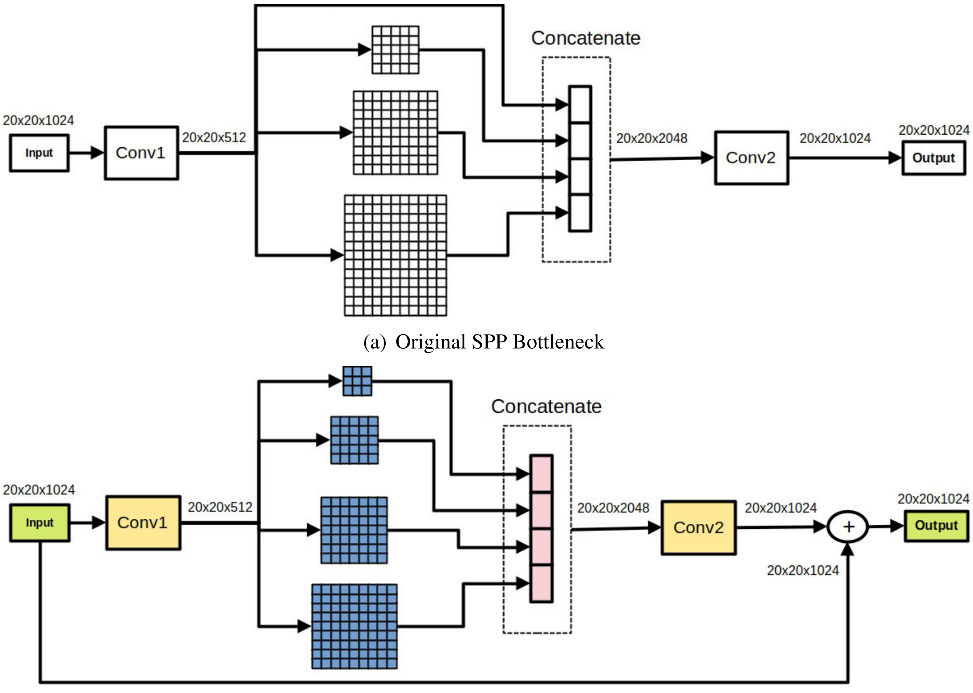
The information is concatenated to get an output feature map of $20 \times 20 \times 2048$ as given in Equation 13. We propose an improved SPP module by enhancing the receptive field of the input feature map. These parallel layers are characterized by different smaller kernel sizes, stride, and padding detailed in Table 2. This aggregation of features [37] using smaller to larger kernel sizes tends to retain feature details which are further preserved using the concatenate operation shown in Figure 4. The spatial information gathered as $Y_{i,j,k}$ and the size of the output feature map is $20 \times 20 \times 1024$ after $Conv_2$ operation, the 1×1 convolution is again carried out to return the feature map size to 1024. In order to avoid overfitting due to vanishing gradients, we used an identity skip connection to keep the information at the initial layers and performed matrix addition as shown in Equation 14. Here, $X_{i,j,k}$ is the input tensor of the SPP module and $Y_{i,j,k}$ is the transformed tensor after the convolution operation. Like the default YOLOX model, the input feature map size is preserved. Finally, the aggregated low-level features are fed into the FPN neck block for feature fusion.

$$Y_{i,j,k} = Conv_1\left(cat\left(\begin{array}{l} max^3(Conv_2), \\ max^5(Conv_2), \\ max^7(Conv_2), \\ max^9(Conv_2) \end{array}\right)\right) \tag{13}$$

$$Z_{i,j,k} = \tau(Y_{i.j.k}) + X_{i,j,k} \tag{14}$$

IV. PROPOSED DATASET

Cotton is a Kharif crop sown from March to May in various regions of Sindh and Punjab in Pakistan. For the sake of this research, a small orchid with approximately 50 plants was chosen to capture images. To keep the conditions favorable for the pathogens to grow, no additional pesticide or fertilizer was used. The images were captured using a smartphone (Samsung A21) kept $20 - 30$ cm away from the leaf at varying angles. Images were captured under varying lighting conditions in the afternoon in both sunny and cloudy weather. The focus of the smartphone was adjusted to capture a single leaf only. A total of 464 images were collected and annotated with the help of expert pathologists. The data was divided into



(a) Original SPP Bottleneck

(b) Improved SPP Bottleneck

FIGURE 4. Comparison of original Spatial Pyramid Pooling (SPP) module with our proposed SPP module of YOLOX.

TABLE 3. Details of Cotton disease severity dataset before augmentation.

Disease Type & Stage	Number of Images
Healthy	140
Curl Stage-1	70
Curl Stage-2	65
Curl Stage-1 + Sooty Mold	51
Curl Stage-2 + Sooty Mold	90
Leaf Enation + Curl Stage-2	26
Curl Stage-1 + Curl Stage-2 + Sooty Mold	22
Total	464

seven types of cotton curl severity and disease coexistence images. The detail of the dataset before applying augmentation is mentioned in Table 3. Some sample images taken from the dataset are shown in Figure 5.

Cotton curl is a common disease found in Pakistan and India. It is caused by begomoviruses and the spread is influenced by white flies. Early symptoms are leaf chlorosis, labeled as stage-1 of cotton leaf curl. Within 2-3 weeks, the symptoms worsen with vein-thickening along with darkening of the leaf starts. Vein thickening can be clearly observed from the backside of the leaf. It is followed by the curling of leaves around its edges and is labeled as stage-2 of the disease. Three weeks later, dark black specks start appearing on the leaves hindering the spread of the virus. The pathologists identify it as cotton sooty mold caused by the spread of aphids. Almost all the plants in the vicinity get infected with the curl virus and sooty mold. Images in this class are captured from ad-axial and ab-axial surfaces as symptoms of vein thinking and sooty mold appear on both sides and making the dataset more generalized. Soon as the symptoms of sooty mold subside, leaf enations start appearing, labeled

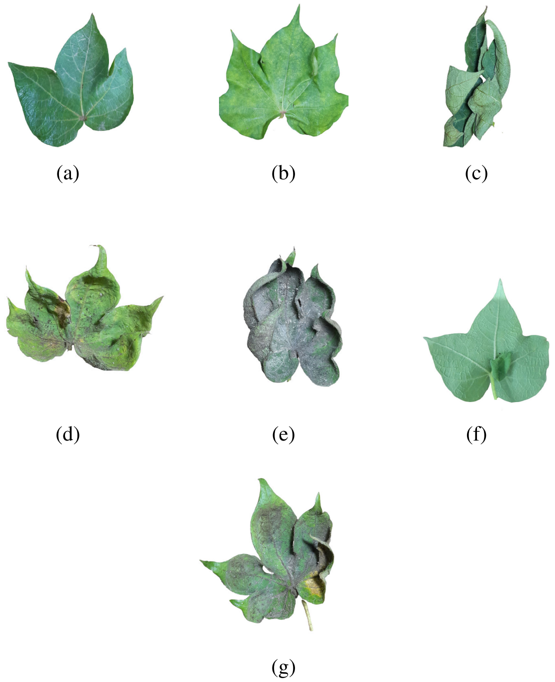


FIGURE 5. Sample images taken from the dataset (a) Healthy (b) Curl stage-1 (c) Curl stage-2 (d) Curl stage-1 + Sooty mold (e) Curl stage-2 + Sooty mold (f) Leaf enation + Curl stage-2 (g) Curl stage-1 + Curl stage-2 + Sooty mold.

as leaf enation. Since leaf enation always appears on the underside of the leaf, only abaxial images are captured in this class.

A. IMAGE ANNOTATION AND AUGMENTATION

Ground truth boxes were manually drawn using the Roboflow annotation tool. Before marking the ground truth boxes the labels of diseased regions were carefully marked with the help of pathologists. The rectangular bounding boxes were manually marked with their corresponding labels by avoiding null regions. Several boxes have overlapping edges due to the existence of overlapping symptoms. Around 80% of the image set is used for training & validation and the rest was used for testing. Images were imported in the model input in Pascal VOC format which contains image localization and annotation details in an XML file. The file contains the bounding box coordinates, height, width, and class label for each box as shown in the Figure 6. For each image there is an associated XML file when imported into the model for training, validation or testing [17].

The dataset used for experiments contain images captured at varying angles and sizes. Hence, we performed an auto-orient operation to keep the bounding boxes on the object of interest irrespective of any augmentation step performed on it later. Further, the images were resized using a normalization set at a resolution of 416 × 416. as commonly required by YOLO. To avoid overfitting while dealing with an imbalanced dataset, we prefer to perform augmentation steps of flip horizontal, 90° rotation, and brightness(±25%). Applying these augmentations will not only enhance the size of the dataset but will add more diversity to the images captured under different lighting conditions. After randomly applying augmentation the size of the dataset is raised to $1,112$. All these pre-processing steps were performed using Roboflow data services. A snapshot of the initially pre-processed image and after data augmentation is shown in Figure 7.

V. EXPERIMENTS

A. EXPERIMENTAL DETAILS

All experiments were performed on Intel(R) Core(TM) i3-1005G1 1.20GHz CPU and NVIDIA Tesla T4 GPU under Windows 10 operating system. Complete details of the experimental workbench are given in Table 4. The experimental environment was set up after installing the required libraries and repositories. The dataset was imported into the notebook and pretrained weights for the VOC dataset were downloaded for YOLOX-s. Our proposed model was trained using hyperparameters mentioned in Table 5 for 100 epochs. With a batch size of 32, the best weights obtained after training were saved and were further used to test the performance of our proposed model on test images. Images in the dataset were resized to 640 × 640 for the purpose of experimentation. Finally, 80% images were used for training and validation purposes while the remaining 20% were later used for testing.

TABLE 4. A summary of the experimental setup.

Hardware & Software	Details
CPU	Intel(R) Core(TM) i3, 10^{th} generation
Programming Language	Python 3.7
Deep Learning Framework	Pytorch 1.8.0
GPU	NVIDIA Tesla T4
GPU Environment	Google Colab CUDA 11.2

TABLE 5. A summary of hyperparameters used for training.

Training Parameter	Value
Optimizer	SGD
Batch Size	32
Max Epochs	100
Base Learning rate	0.01
Weight decay	0.0005
NMS threshold	0.65
Momentum	0.9
Mixup& Mosaic Augmentation	True
α in Reg-loss	3

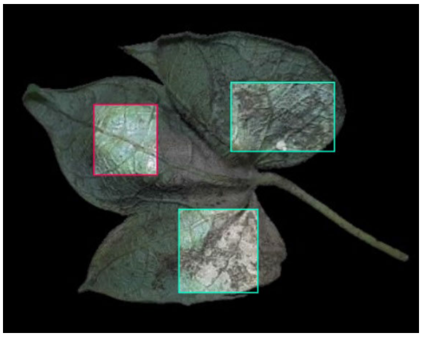
A strong mixup and mosaic augmentation were used for all training epochs. The Stochastic Gradient Descent (SGD) algorithm with a base learning rate of 0.01 is used for optimizing the model performance with a momentum of 0.9 was used and a weight decay of 0.0005. Cosine learning was used as the default learning rate scheduler. The training process was analyzed to conclude model performance based on several metrics. To efficiently differentiate between the overlapping symptoms, the NMS threshold was kept at 0.65. For classification and objectness, we use BCE loss with Logits. Using sigmoid activation instead of class probabilities provides more stable results during inference [38]. For the bounding box, regression loss $\mathcal{L}_\alpha$ was used. The training convergence was also observed and compared with other loss functions discussed in Section VI.

VI. RESULTS & DISCUSSION

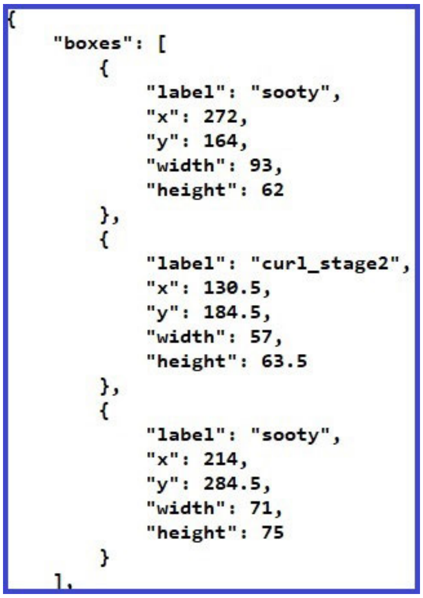
Training performance was monitored for improved convergence speed and detection performance of co-occurring diseased symptoms and severity classes on self-collected datasets. The metric commonly used for detection models is Mean Average Precision (mAP) which shows the precision for all classes when the Intersection over Union (IoU) threshold is 50%. The mathematics of the computation is given in Equation 15, where $i = 1, 2, \cdots, 5$. Even for this work, the mAP score was computed by keeping the IoU threshold at 0.5.

$$mAP = \frac{1}{5}\sum_{i=1}^{5} AP_i \tag{15}$$

The default YOLOX model showed comparable training accuracy in the beginning but it failed to converge in the last 30 epochs as shown in Figure 8. This led to lower accuracy of the default YOLOX model (69.90%) compared to the proposed improved YOLOX model (73.13%) as shown in



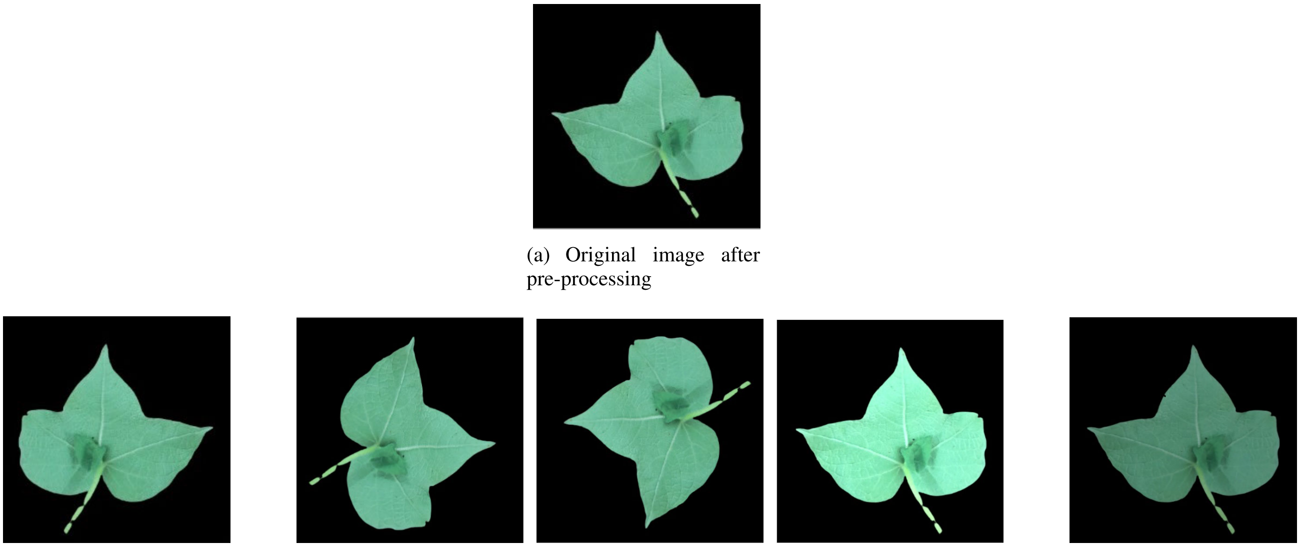
(a) Annotated image with bounding boxes



```
{
    "boxes": [
        {
            "label": "sooty",
            "x": 272,
            "y": 164,
            "width": 93,
            "height": 62
        },
        {
            "label": "curl_stage2",
            "x": 130.5,
            "y": 184.5,
            "width": 57,
            "height": 63.5
        },
        {
            "label": "sooty",
            "x": 214,
            "y": 284.5,
            "width": 71,
            "height": 75
        }
    ],
```

(b) XML file containing annotation information

FIGURE 6. Annotation of Cotton Severity Dataset.



(a) Original image after pre-processing

(b) Flip Horizontal

(c) 90°Rotate clockwise

(d) 90°Rotate counter clockwise

(e) Brightness +25%

(f) Brightness−25%

FIGURE 7. Original image and its augmented version.

Figure 8. In an attempt to improve the detection performance, authors also experimented by adding an SE block in the SPP module of default YOLOX [39]. The training performance for the YOLOX-SE model clearly got overfit in the last 20 training epochs with an inferior inference speed as shown in Table 6. The inference time taken by the default YOLOX is slightly better than our model but misclassifications were detected for several classes.

The validation and test results of the model were evaluated on various IOU losses to achieve the best convergence and mAP performance. The model achieved the best results on improved SPP block with $\mathcal{L}_{CIoU}$ with modulating factor $\alpha = 3$.

Further, a decrease in total loss during the training process is shown in Figure 9. Total loss is the average of all losses as shown in Equation 16.

$$Loss_{total} = Loss_{cls} + Loss_{reg} + Loss_{obj} \tag{16}$$

where classification loss $Loss_{cls}$ and Objectness loss $Loss_{obj}$ evaluated by BCE loss and $Loss_{reg}$ is the regression loss.

As the training epochs progress, the loss decreases rapidly and then converges gradually. The performance of $\mathcal{L}_{\alpha-IoU}$ is

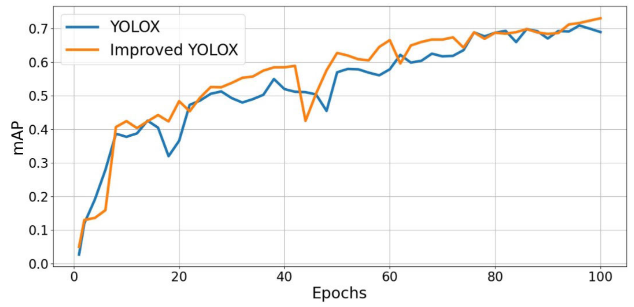


FIGURE 8. A comparison of mAP training curve of Improved and the default YOLOX models during training.

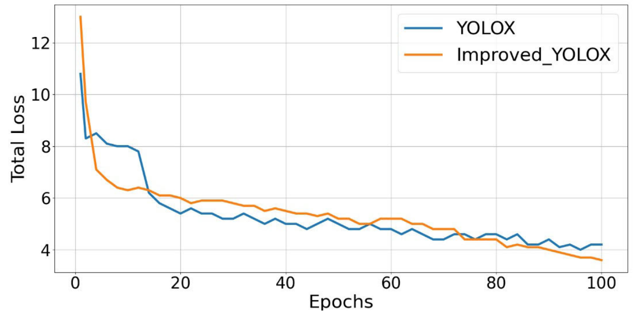


FIGURE 9. A comparison of the total loss of the Improved and the default YOLOX models during training.

better than vanilla-IoU and other regression techniques for the required localization of co-occurring disease symptoms.

A. ABLATION STUDY

Our proposed model uses an improved SPP block that integrates multilevel spatial features using a Maxpool network with smaller kernel sizes. Which extracts the useful feature information more effectively; the training mAP achieved was 71.12% which is better than the default SPP which uses $\mathcal{L}_{IoU}$ as bounding box loss.

To obtain the optimal training performance of the proposed improved YOLOX model, several loss functions were examined to refine the position of the predicted bounding box. $\mathcal{L}_{CIoU}$, $\mathcal{L}_{EIoU}$, $\mathcal{L}_{DIoU}$, $\mathcal{L}_{\alpha-IoU}$ and $\mathcal{L}_{IoU}$ are compared on the basis of the detection accuracy and training time per epoch as shown in Figure 10. Detection accuracy values obtained using vanilla IoU and GIoU were found comparable but the latter is slower to converge. EIoU turned out to be the slowest with a comparatively inferior value of mAP and hence the worst. The proposed alpha IoU performed the best with the largest value of mAP and a time to converge value just greater than the one we achieved for GIoU. To investigate the effect of modulating factor α we compared the performance of our model with varying values of α for $\mathcal{L}_{\alpha-IoU}$. The relative performance of $\mathcal{L}_{\alpha-IoU}$ was analyzed by varying the value of modulating factor α. Experimentation revealed that regression accuracy tends to increase as the value of α increases. We concluded that better detection performance was achieved by setting the $\alpha = 3$ for our specific dataset as shown in Figure 11. The best results were shown on our improved YOLOx model using the $\mathcal{L}_{\alpha-IoU}$ as bounding box regression loss as shown in Table 7.

B. DETECTION PERFORMANCE ON TEST DATASET

Once the training is complete; best weights are obtained which are then used to evaluate the performance of the model on the test dataset. The regression task is complete when the objectness of the bounding box is evaluated. The BCE loss is used to indicate the presence of an object within a bounding box. The objectness of the presence of disease in the predicted bounding box is evaluated based on the confidence score. Equation 17 gives a way to compute conditional class score C_{CS} given the class confidence score and the conditional class probability P_{ic}.

$$C_{CS} = Box\_confidence\_score \times P_{ic} \tag{17}$$

In order to achieve a better detection mAP score we experimented to select an optimal threshold value, which should not be low to give a false positive and high enough to miss true predictions. Hence, after evaluating the test results the confidence threshold was kept at 0.25. Figure 12 shows the detection performance of Improved YOLOX on the test dataset. Each predicted bounding box marks a confidence score in the test images. The score shows confidence in both classification and localization.

YOLOX model uses the same baseline model using SPP anchor-free mechanism. Also, the decoupled head helps reducing the convergence speed which was observed while comparing the performance with YOLOv4 [40] and YOLOv5 [20] models that were also trained and validated on our Cotton severity dataset and the results based on precision and inference time are also shown in Table 8. It can be seen clearly that although the inference speed of YOLOv4 and YOLOv5 is much better than our model the detection accuracy is compromised. The anchor-free mechanism reduces the complexity and probable bottlenecks during training hence the default YOLOX model and our Improved YOLOX models show better convergence performance than other models trained on our dataset. Figure 8 shows values of mAP, achieved during training using the default YOLOX and the proposed improved YOLOX models.

To analyze the performance of our proposed YOLOX model to classify severity stages and overlapping symptoms. Curl stage-1 was detected with relatively lower average precision because it was misclassified with the healthy and curl stage-2 class. Similarly, the overlapping symptoms of sooty mold and curl stage-2 were detected with better precision with our improved YOLOX model. Figure 13 shows the confusion matrix results of the Improved YOLOX model compared with the original YOLOX on test data. The models accurately detect leaf enation as the class is not visually similar to any other classes and images in our test dataset have not many challenging images for the class. The healthy class was also misclassified with the curl stage-1 because of the similarity in visual symptoms. Similarly, sooty mold

TABLE 6. A comprehensive summary of results in comparison with other models on test data.

Models	Input Size	$mAP@0.5$	AP (Curl stage-1)	AP (Curl stage-2)	AP (Healthy)	AP (Leaf Enation)	AP (Sooty)	Inference Time
YOLOv4	416×416	47.42	56.61	40.01	27.99	58.12	55.98	20.79
YOLOv5	416×416	53.11	29.10	47.05	66.80	70.51	52.13	15.00
YOLOX	640×640	69.04	49.65	62.12	71.20	100	62.17	31.10
YOLOX-SE	640×640	66.66	65.66	63.55	55.68	88.66	59.75	56.87
Improved YOLOX	640×640	72.31	60.22	65.76	61.55	100	74.02	32.03

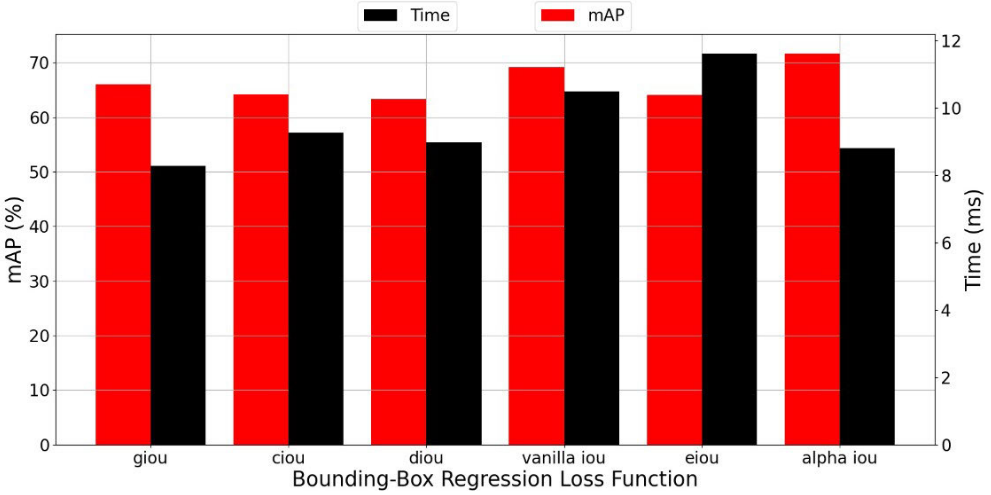


FIGURE 10. Comparison of various bounding box regression techniques on the basis of mAP and average time per epoch.

TABLE 7. Results of an ablation study conducted during training the model with varying strategies.

SPP (5,9,13)	SPP (3, 5, 7, 9)	$\mathcal{L}_{IoU}$	$\mathcal{L}_{\alpha-IoU}$	Skip Connection	mAP %
√	×	√	×	×	69.90
×	√	√	×	×	67.51
×	√	√	×	√	71.12
×	√	×	√	√	**73.13**

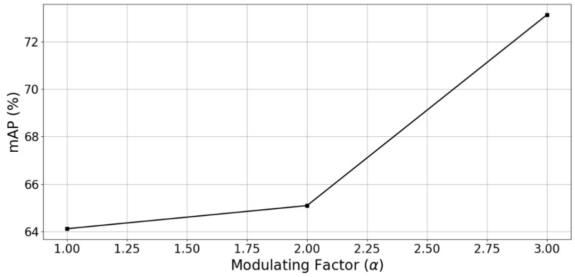


FIGURE 11. Performance of proposed YOLOX model using $\mathcal{L}_{\alpha-IoU}$ as bounding box regression loss with varying values of α.

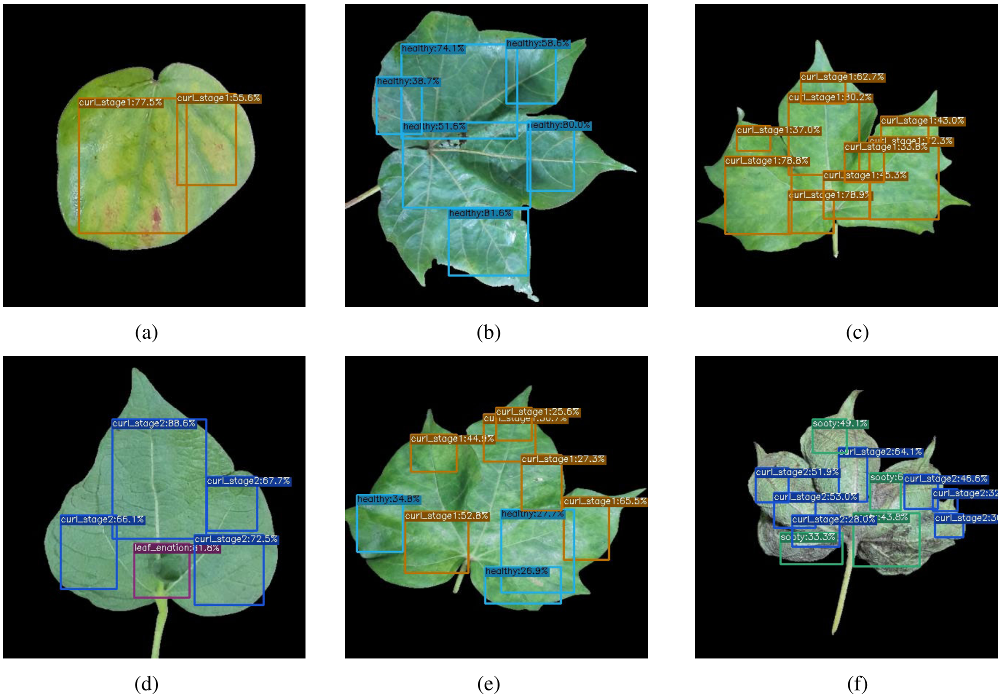
TABLE 8. A comparison of the performance of our proposed model with state-of-the-art models.

Model	Training mAP (%)
YOLOv4[40]	50.12
YOLOv5[20]	58.64
YOLOv7 [41]	35.0
EfficientDet [42]	32.11
YOLOX-ti-lite [26]	62.89
Our proposed Technique	**73.13**

symptoms were falsely detected as curl stage 1 and curl stage 2 because of their overlapping characteristics as shown in Figure 13(a). Curl stage 1 and sooty mold were correctly classified with better precision by our proposed model as shown in Figure 13(b). However, the healthy class was misclassified as curl stage 1.

C. COMPARISON WITH STATE-OF-THE-ART TECHNIQUES

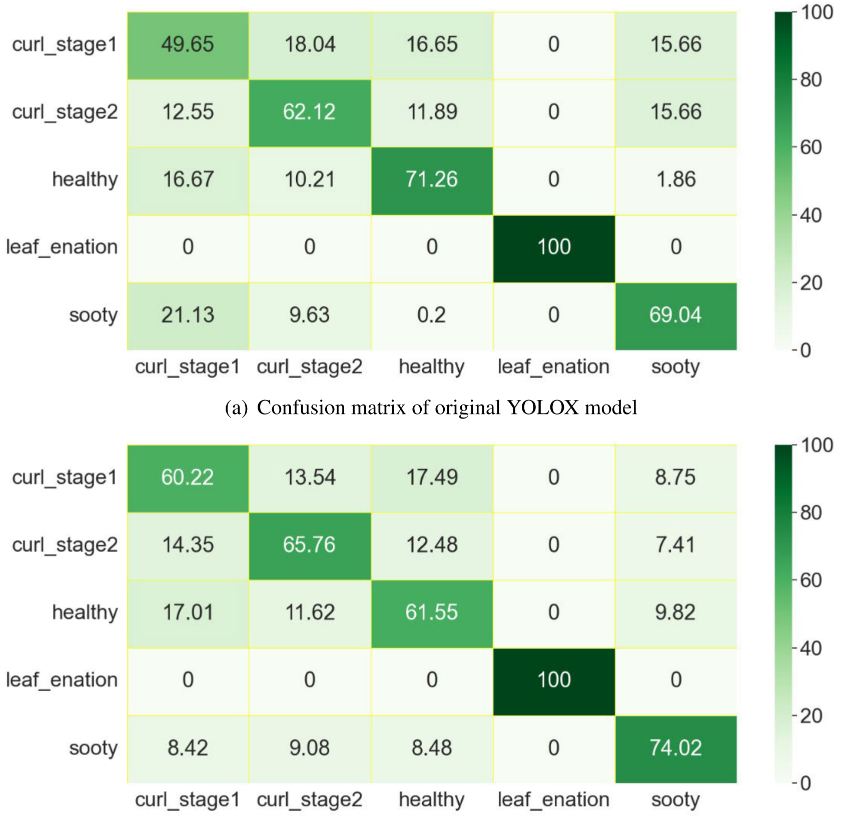
In this section, we compare the performance of our proposed improved YOLOX model with other state-of-the-art models for our dataset in terms of training accuracy. We used the default code settings for training YOLOv4, YOLOv5, YOLOv7, Efficientdet, and YOLOX-ti-lite on our cotton severity dataset for 100 epochs. The mAP scores are significantly low as compared to our proposed YOLOX model as shown in Table 8. We also trained the YOLOX-ti-lite model which is a version of YOLOX preferably used for edge devices. The model has an optimized SPP block with efficient performance on embedded devices. But it failed to give desirable results for a specific dataset.



(a) (b) (c)

(d) (e) (f)

FIGURE 12. Test results obtained using our proposed technique.



(a) Confusion matrix of original YOLOX model

(b) Confusion matrix of our proposed technique

FIGURE 13. Confusion matrices on five disease symptoms using the original YOLOX model and our proposed technique.

VII. CONCLUSION

The proposed work proposes a framework to detect multiple diseases on a single leaf and identifying progressive severity symptoms of a single disease on cotton plants. For this, we have proposed YOLOX based model with an improved Spatial Pyramid Pooling block. Multi-scale features were aggregated with the help of diverse pooling rates. Spatial information was further preserved by introducing residual connections. With this model, we achieved better identification results in the case of similar-looking and overlapping disease symptoms. Experimental results showed that the proposed model achieved 73.13% mAP on training and 72.31% mAP on the test dataset which is 3.27% better than the default YOLOX. Further, the best results for overlapping and co-existing classes like Curl stage-2 and Sooty mold were obtained with an average precision of 65.76% and 74.02% respectively. Better generalization and loss convergence were achieved using $\mathcal{L}_{\alpha-CIoU}$ as bounding box regression loss.

VIII. LIMITATIONS & FUTURE WORK

The presence of multiple stresses on a single host and the appearance of varying stages of diseases on a single leaf are two commonly faced situations in the field. This work proposes a deep learning-based solution for these two scenarios that has successfully been tested on a rich dataset. However, to further improve it in the future, the following are some future directions:

1) We aspire to make our model robust and suitable for real-field practical applications. In order to make the dataset more widely applicable, efforts will be made in the future to expand it by collecting more images pertaining to other diseases and crops. Moreover, the proposed model is trained and tested on a single leaf in a clean background. To enhance its practicability, we will invest in adding more real field samples.
2) To deploy the trained model on edge devices, we will work on making the design lightweight in order to accelerate its training and testing speed.
3) Much work is required to improve the precision and confidence score for challenging real-field plant diseased images by exploring the fusion of low-level spatial features without compromising on the model's complexity.

ACKNOWLEDGMENT

The authors would like to thank the support provided by the Department of Plant Pathology, The Islamia University of Bahawalpur, Pakistan. They would also like to thank the technical support provided by Dr. Naveed Aslam and Dr. Anam Moosa in identifying, annotating, and marking severity stages for various plant diseases.

REFERENCES

[1] J. G. A. Barbedo, "Factors influencing the use of deep learning for plant disease recognition," *Biosyst. Eng.*, vol. 172, pp. 84–91, Aug. 2018.

[2] Z. Lin, S. Mu, F. Huang, K. A. Mateen, M. Wang, W. Gao, and J. Jia, "A unified matrix-based convolutional neural network for fine-grained image classification of wheat leaf diseases," *IEEE Access*, vol. 7, pp. 11570–11590, 2019.

[3] M. H. Saleem, S. Khanchi, J. Potgieter, and K. M. Arif, "Image-based plant disease identification by deep learning meta-architectures," *Plants*, vol. 9, no. 11, p. 1451, Oct. 2020.

[4] R. Wang, L. Jiao, C. Xie, P. Chen, J. Du, and R. Li, "S-RPN: Sampling-balanced region proposal network for small crop pest detection," *Comput. Electron. Agricult.*, vol. 187, Aug. 2021, Art. no. 106290.

[5] D. Jiang, G. Li, C. Tan, L. Huang, Y. Sun, and J. Kong, "Semantic segmentation for multiscale target based on object recognition using the improved Faster-RCNN model," *Future Gener. Comput. Syst.*, vol. 123, pp. 94–104, Oct. 2021.

[6] 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*, vol. 17, no. 9, p. 2022, 2017.

[7] N. Kundu, G. Rani, V. S. Dhaka, K. Gupta, S. C. Nayak, S. Verma, M. F. Ijaz, and M. Woźniak, "IoT and interpretable machine learning based framework for disease prediction in pearl millet," *Sensors*, vol. 21, no. 16, p. 5386, Aug. 2021.

[8] S. K. Upadhyay and A. Kumar, "A novel approach for Rice plant diseases classification with deep convolutional neural network," *Int. J. Inf. Technol.*, vol. 14, no. 1, pp. 185–199, Feb. 2022.

[9] B. M. Patil and V. Burkpalli, "Segmentation of cotton leaf images using a modified chan vese method," *Multimedia Tools Appl.*, vol. 81, no. 11, pp. 15419–15437, May 2022.

[10] O. O. Abayomi-Alli, R. Damaševičius, S. Misra, and R. Maskeliūnas, "Cassava disease recognition from low-quality images using enhanced data augmentation model and deep learning," *Exp. Syst.*, vol. 38, no. 7, 2021, Art. no. e12746.

[11] A. Almadhor, H. T. Rauf, M. I. U. Lali, R. Damaševičius, B. Alouffi, and A. Alharbi, "AI-driven framework for recognition of guava plant diseases through machine learning from DSLR camera sensor based high resolution imagery," *Sensors*, vol. 21, no. 11, p. 3830, 2021.

[12] A. Pramanik, S. K. Pal, J. Maiti, and P. Mitra, "Granulated RCNN and multi-class deep SORT for multi-object detection and tracking," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 6, no. 1, pp. 171–181, Feb. 2022.

[13] G. Zhou, W. Zhang, A. Chen, M. He, and X. Ma, "Rapid detection of Rice disease based on FCM-KM and faster R-CNN fusion," *IEEE Access*, vol. 7, pp. 143190–143206, 2019.

[14] Z. U. Rehman, M. A. Khan, F. Ahmed, R. Damaševičius, S. R. Naqvi, W. Nisar, and K. Javed, "Recognizing apple leaf diseases using a novel parallel real-time processing framework based on mask RCNN and transfer learning: An application for smart agriculture," *IET Image Processing*, vol. 15, no. 10, pp. 2157–2168, 2021.

[15] D. Li, R. Wang, C. Xie, L. Liu, J. Zhang, R. Li, F. Wang, M. Zhou, and W. Liu, "A recognition method for Rice plant diseases and pests video detection based on deep convolutional neural network," *Sensors*, vol. 20, no. 3, p. 578, Jan. 2020.

[16] M. H. Saleem, J. Potgieter, and K. M. Arif, "Weed detection by faster RCNN model: An enhanced anchor box approach," *Agronomy*, vol. 12, no. 7, p. 1580, Jun. 2022.

[17] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang, "Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks," *IEEE Access*, vol. 7, pp. 59069–59080, 2019.

[18] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[19] X. Wang and J. Liu, "Tomato anomalies detection in greenhouse scenarios based on YOLO-dense," *Frontiers Plant Sci.*, vol. 12, p. 533, Apr. 2021.

[20] M. P. Mathew and T. Y. Mahesh, "Leaf-based disease detection in bell pepper plant using YOLO v5," *Signal, Image Video Process.*, vol. 16, no. 3, pp. 841–847, Apr. 2022.

[21] J. Yao, Y. Wang, Y. Xiang, J. Yang, Y. Zhu, X. Li, S. Li, J. Zhang, and G. Gong, "Two-stage detection algorithm for kiwifruit leaf diseases based on deep learning," *Plants*, vol. 11, no. 6, p. 768, Mar. 2022.

[22] Z. Chen, R. Wu, Y. Lin, C. Li, S. Chen, Z. Yuan, S. Chen, and X. Zou, "Plant disease recognition model based on improved YOLOv5," *Agronomy*, vol. 12, no. 2, p. 365, Jan. 2022.

[23] G. Ganesan and J. Chinnappan, "Hybridization of ResNet with YOLO classifier for automated paddy leaf disease recognition: An optimized model," *J. Field Robot.*, vol. 39, no. 7, pp. 1085–1109, Oct. 2022.

[24] J. Pan, L. Xia, Q. Wu, Y. Guo, Y. Chen, and X. Tian, "Automatic strawberry leaf scorch severity estimation via Faster R-CNN and few-shot learning," *Ecol. Informat.*, vol. 70, Sep. 2022, Art. no. 101706.

[25] P. Zhang, L. Yang, and D. Li, "EfficientNet-B4-Ranger: A novel method for greenhouse cucumber disease recognition under natural complex environment," *Comput. Electron. Agricult.*, vol. 176, Sep. 2020, Art. no. 105652.

[26] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO series in 2021," 2021, *arXiv:2107.08430*.

[27] I. Pacal, A. Karaman, D. Karaboga, B. Akay, A. Basturk, U. Nalbantoglu, and S. Coskun, "An efficient real-time colonic polyp detection with YOLO algorithms trained by using negative samples and large datasets," *Comput. Biol. Med.*, vol. 141, Feb. 2022, Art. no. 105031.

[28] H. Zhai, J. Cheng, and M. Wang, "Rethink the IoU-based loss functions for bounding box regression," in *Proc. IEEE 9th Joint Int. Inf. Technol. Artif. Intell. Conf. (ITAIC)*, Dec. 2020, pp. 1522–1528.

[29] S. K. Noon, M. Amjad, M. A. Qureshi, and A. Mannan, "Use of deep learning techniques for identification of plant leaf stresses: A review," *Sustain. Comput., Informat. Syst.*, vol. 28, Dec. 2020, Art. no. 100443.

[30] S. M. Jaisakthi, P. Mirunalini, D. Thenmozhi, and Vatsala, "Grape leaf disease identification using machine learning techniques," in *Proc. Int. Conf. Comput. Intell. Data Sci. (ICCIDS)*, Feb. 2019, pp. 1–6.

[31] S. Sun, M. Jiang, D. He, Y. Long, and H. Song, "Recognition of green apples in an orchard environment by combining the GrabCut model and Ncut algorithm," *Biosyst. Eng.*, vol. 187, pp. 201–213, Nov. 2019.

[32] Y. Li, Z. Guo, F. Shuang, M. Zhang, and X. Li, "Key technologies of machine vision for weeding robots: A review and benchmark," *Comput. Electron. Agricult.*, vol. 196, May 2022, Art. no. 106880.

[33] G. Wang, H. Zheng, and X. Zhang, "A robust checkerboard corner detection method for camera calibration based on improved YOLOX," *Frontiers Phys.*, vol. 9, p. 828, Feb. 2022.

[34] U. Ruby and V. Yendapalli, "Binary cross entropy with deep learning technique for image classification," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, no. 4, pp. 5393–5397, Aug. 2020.

[35] Z. Gevorgyan, "SIoU loss: More powerful learning for bounding box regression," 2022, *arXiv:2205.12740*.

[36] J. He, S. Erfani, X. Ma, J. Bailey, Y. Chi, and X.-S. Hua, "α-IoU: A family of power intersection over union losses for bounding box regression," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 20230–20242.

[37] C.-L. Wang, M.-W. Li, Y.-K. Chan, S.-S. Yu, J. H. Ou, C.-Y. Chen, M.-H. Lee, and C.-H. Lin, "Multi-scale features fusion convolutional neural networks for Rice leaf disease identification," *J. Imag. Sci. Technol.*, vol. 66, no. 5, pp. 1–12, 2022.

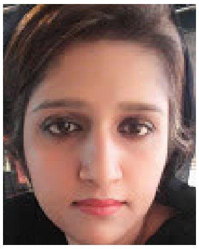
[38] T. Wu, Q. Huang, Z. Liu, Y. Wang, and D. Lin, "Distribution-balanced loss for multi-label classification in long-tailed datasets," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 162–178.

[39] G. Li, X. Huang, J. Ai, Z. Yi, and W. Xie, "Lemon-YOLO: An efficient object detection method for lemons in the natural environment," *IET Image Process.*, vol. 15, no. 9, pp. 1998–2009, Jul. 2021.

[40] A. Bochkovskiy, C.-Y. Wang, and H.-Y. Mark Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[41] C.-Y. Wang, A. Bochkovskiy, and H.-Y. Mark Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 2022, *arXiv:2207.02696*.

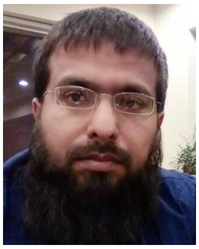
[42] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10781–10790.



SEROSH KARIM NOON (Graduate Student Member, IEEE) received the B.Sc. degree in electronics engineering from the NFC Institute of Engineering and Technology, Multan, Pakistan, in 2006, and the M.S. degree in electrical engineering from the University of Engineering and Technology, Lahore. She is currently pursuing the Ph.D. degree in electrical engineering with The Islamia University of Bahawalpur, Pakistan. She is also working as an Assistant Professor with the Department of Electrical Engineering, NFC Institute of Engineering and Technology. Her research interests include computer vision and machine learning.

MUHAMMAD AMJAD received the B.Sc. and M.Sc. degrees in electrical engineering from the University of Engineering and Technology, Lahore, in 1998 and 2005, respectively, and the Ph.D. degree in electrical engineering from Universiti Teknologi Malaysia, in 2013. He is currently working as a Professor and the Dean of the Faculty of Engineering, The Islamia University of Bahawalpur, Pakistan. His research interests include power electronics and smart grid.

MUHAMMAD ALI QURESHI (Senior Member, IEEE) received the B.Sc. degree in electrical engineering from the University of Engineering and Technology, Lahore, Pakistan, in 2000, the M.Sc. degree in telecommunication engineering from NWFP-UET Peshawar, Pakistan, in 2008, and the Ph.D. degree in electrical engineering from the King Fahd University of Petroleum and Minerals (KFUPM), Saudi Arabia, in 2017. He is currently working as an Associate Professor and the Chairperson of the Department of Information and Communication Engineering, The Islamia University of Bahawalpur, Pakistan. His research interests include image and video processing, image forensics, and image quality assessment. He is the Chair of the IEEE Bahawalpur Subsection Region 10.



ABDUL MANNAN received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from the University of Engineering and Technology, (UET) Lahore, in 2002, 2007, and 2021, respectively. He is currently with the Department of Biomedical Engineering, NFC Institute of Engineering and Technology, Multan, Pakistan. His research interests include computer vision, image processing, and machine learning.