Received: May 12, 2023. Revised: June 13, 2023.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

An Effective Data Augmentation Based on Uncertainty Based Progressive Conditional Generative Adversarial Network for improving Plant Leaf Disease Classification

Balaji Sreenivasulu[1*] **Anjaneyulu Pasala[2]** **Gaikwad Vasanth[3]**

[1]*Department of Computer Science and Engineering, Bangalore Technological Institute, Bangalore, India*
[2]*Department of Artificial Intelligence and Data Science (AI&DS), CMR Institute of Technology, Bangalore, India*
[3]*Department of Computer Science and Engineering, Government Engineering College Ramanagara, India*
* Corresponding author's Email: balajichetty6@gmail.com

Abstract: Early discovery and precise classification of plant-lead diseases are used to handle their spread and enhance the overall yield and product quality. The deep learning approach attains significant achievements in image classification and recognition. Since the classification using deep learning mainly depend on large-scale dataset for preventing the overfitting issue. Image augmentation is required to be developed to eliminate the risk of overfitting during the classification. In this research, the deep learning model based augmentation namely uncertainty based progressive conditional generative adversarial network (UPC-GAN) is developed for improving the plant leaf disease classification. The UPCGAN is used to map the images from one domain to another domain in a paired manner and estimate the uncertainty with the created images. Moreover, UPCGAN performs pixel wise residual distribution using the independent distributed zero mean generalized Gaussian distribution (GGD). The progressive learning of UPC-GAN increases the differences in the augmented synthetic images for improving the classification using DenseNet121. The dataset used to evaluate the proposed UPCGAN-DenseNet121 method is PlantVillage dataset. The performance of UPCGAN-DenseNet121 is analysed using accuracy, precision, recall and F1-score. Existing research such as deep convolutional GAN (DCGAN)-GoogleNet, conditional GAN (CGAN)-DenseNet121 and Fast wide and deep feature extraction block (WDBlock) based GAN namely FWDGAN are used to evaluate the UPCGAN-DenseNet121. The accuracy of UPCGAN-DenseNet121 for 10 classes is 98.2%, which is high when compared to the DCGAN-GoogleNet, CGAN-DenseNet121 and FWDGAN.

Keywords: Deep learning model-based augmentation, Densenet121, Uncertainty based progressive conditional generative adversarial network, Plant leaf disease classification, Synthetic images.

1. Introduction

One of the significant characteristics of precision agriculture is dealing with plant disease. Plants are habitually suffered from various unknown diseases that decrease the overall yield and reduce production quantity and quality [1]. Plant disease creates a danger to global food security and smallholder farmers whose life mainly depends on healthy crop and agriculture. In developing countries, smallholder farmers create more than 80% of agricultural production, but statistics state that more than 50% of loss occurred because of pests and diseases. Subsequently, the overall world population is predicted to develop to more than 9.7 billion in the year 2050, hence developing food security is a higher concern in upcoming years [2]. The precise and timely plant disease diagnosis helps for supportable and accurate agriculture and is also used to prevent unwanted waste of financial and other resources [3] [4, 5]. On the other hand, imprecise disease calculations create faulty decisions or possibly return severe issues [6].

The discovery of plant disease using the naked eye is difficult for farmers and locating the expertise in plant disease is also a challenging task on the rural side [7, 8]. The pathogens are controlled when the diseases are detected in a preliminary stage at plants. Some of the issues which frequently occurred during leaf disease classification are brightness, complex image background, pose, interclass similarities, color, and occlusion. Moreover, the initial identification of plant disease is frequently impossible in various parts of the world, because of insufficient resources [9]. Therefore, artificial intelligence is used in agriculture for gaining insights about crops and using the information to maximize overall production [10]. But the small or inadequate data frequently creates overfitting in the models and affects the performances during classification. This issue is eliminated by performing the data augmentation and annotations using a property of Deep Neural Network (DNN) namely transfer learning [11, 12, 13, 14]. Additionally, the DNN can use the raw data directly without any hand crafted features [15]. The general image augmentation approaches such as shift, zoom, and rotation are employed in existing research for allowing to create the new images from the original dataset and maximizing the number of images, but they cannot reduce the misclassifications. Moreover, the existing approaches based on tomato leaf disease was not efficient and leads to misclassification with less accuracy. The aforementioned issue is taken as the motivation for this research to develop an effective augmentation approach for improving classification performance. This research proposed UPCGAN which can improvise the classification efficiency by augmenting the image using a pixel wise residual approach. Moreover, the suggested approach has the tendency to learn and predict the optimal scale and shape of every pixels of the leaf image.

The key contributions are concise as follows:

- The UPCGAN based deep learning data augmentation is developed for generating effective augmented images in various resolutions. The progressive learning of UPCGAN considers the lower to higher resolutions of original images for improving the classification of plant leaf disease.
- Further, the augmented images from UPCGAN and original images are used in the DenseNet121 to classify the plant leaf images. The DenseNet121 is considered in this research because it uses a higher amount of fully connected layers for obtaining a better representation of deeply hidden features.

The paper is arranged as follows: section 2 provides the related research about the data augmentation techniques developed for plant disease detection. A detailed explanation of UPCGAN-DenseNet121 is provided in section 3 whereas the outcomes of the UPCGAN-DenseNet121 are given in section 4. Further, the conclusion is made in section 5.

2. Related work

Pandian [16] developed the 14-layered deep convolutional neural network (14-DCNN) for detecting plant leaf diseases. The data augmentation approaches such as basic image manipulation, deep convolutional GAN, and neural style transfer were used for balancing the individual sizes of each class in the dataset. Here, appropriate hyperparameter values were chosen by using the random search with the coarse to a fine searching approach which was utilized to enhance the training performance of 14-DCNN. However, the occlusion region during image segmentation was not considered which diminish the rate of accuracy during classification.

Wu [17] implemented GAN-based data augmentation for enhancing the accuracy of tomato leaf disease classification. The deep convolutional GAN (DCGAN) was developed to generate the augmented images and GoogleNet was used for the disease prediction. The learning rate, batch size and momentum for generating more realistic and diverse samples were used to optimize the DCGAN. However, imbalance in efficiency was noticed due to usage of noise-to-image GANs which exhibit the image of healthy leaves as diseased leaves.

Zhou [18] presented the fine-grained-GAN to perform local spot area data augmentation. The capacity of spot feature representation was achieved by using hierarchical mask generation in the grape leaf spot data augmentation that was used to enhance the detection of grape leaf spots. An enhanced rapid R-CNN was combined with fine-grained-GAN with a fixed size bounding box which was used to decrease the computations and avoided the scale variation that occurred by the classifier. But, the suggested approach was suited to detect only the visual leaf spots.

Deng [19] developed the RAHC_GAN for increasing the tomato leaf data and discovering diseases. The continuous hidden variables were included in the generator input for controlling the occurred disease area size and for supplementing the intra-class data of the identical disease. For enhancing the concentration of the disease region, a residual attention block was incorporated into the generator. Next, the texture of created image was enriched by using a multi-scale discriminator. Further, ResNet, VGGNet, AlexNet, and GoogleNet were

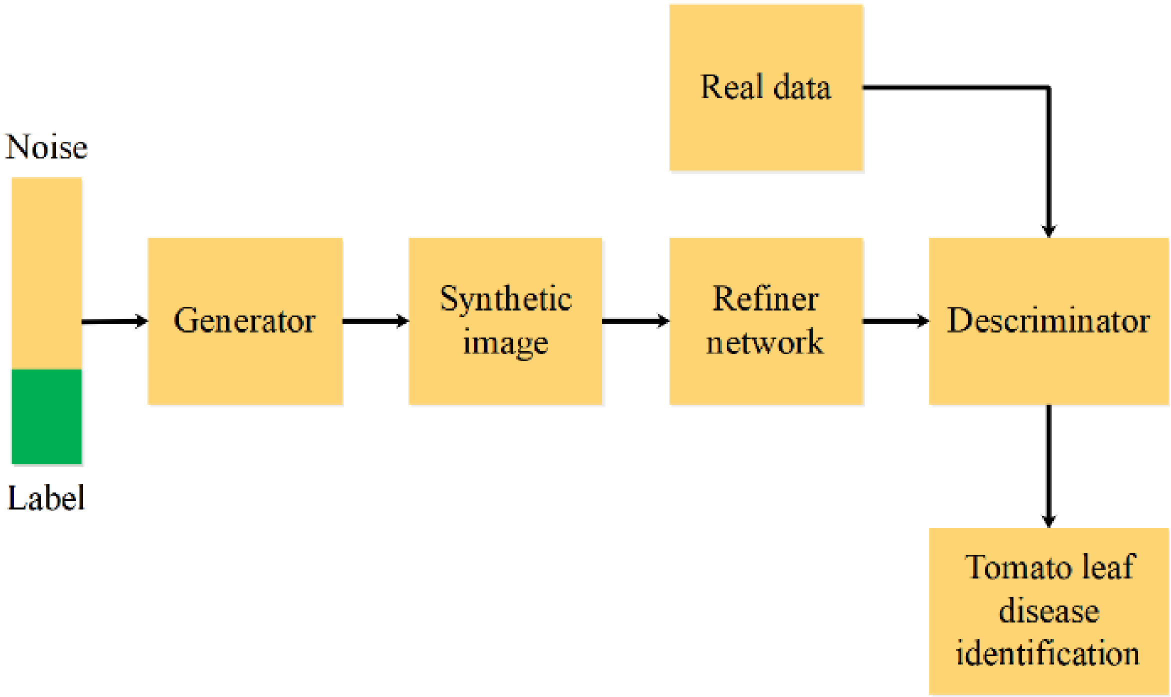

Figure. 1 Block diagram for the overall UPCGAN-DenseNet121 method

used by RAHC_GAN to perform classification. However, GAN creates variability while recognizing the images and diminish the overall efficiency.

Abbas [20] presented the deep learning approach which used the conditional GAN (C-GAN) to create synthetic images for tomato plant leaves. The augmented synthetic images were further used to perform tomato disease identification. Further, the synthetic and real images were used to train the DenseNet121 for classifying tomato leaves diseases. However, the C-GAN was incapable to detect various stages of disease based on its appearance.

Li [21] developed the fast wide and deep feature extraction block (WDBlock) based GAN namely FWDGAN for data augmentation of images. The WDBlock was developed along with two-path strategy for the generator of network. The depth features were obtained using ResNet and global features were obtained using InceptionV1 in WDBlock. The efficiency of augmented leaf disease images was enhanced by using the WDBlock. The Depthwise separable convolution Discriminator was incorporated in discriminator of network for minimizing the model parameters. The developed FWDGAN was required to be analyzed with all classes of tomato leaf disease for an effective prediction.

The issues found from related work are mentioned as follows: accuracy reduction due to lack of image details, variations in the augmented images and inefficiency to detect the different types of diseases. The solutions given by proposed research are stated as follows: A better depiction of deeply hidden features is achieved by using the DenseNet121 with huge amount of fully connected layers. The pixel wise residual dissemination that identify the pixel's optimal scale and shape which results in better classification of UPCGAN-DenseNet121.

3. UPCGAN-DenseNet121 method

In this research, UPCGAN-based data augmentation is accomplished for improving the classification of plant diseases. The overall UPCGAN-DenseNet121 approach is divided into two sections where the UPCGAN generates the synthetic images followed by the classification of plant disease images done using the discriminator. The block diagram for the overall UPCGAN-DenseNet121 method is given in Fig. 1. The input images are given to the generator module where a certain amount of noise is added along with a label for creating the pixel variations. Further, this UPCGAN provides augmented images which leads to improving accuracy.

3.1 Dataset acquisition

In this research, the data is collected from publicly available tomato PlantVillage dataset [22] which is used for evaluating the UPCGAN-DenseNet121 method. This PlantVillage dataset has 16,012 leaf images where it has 10 different classes. In that 10 classes, 9 classes are from diseases and remaining one is healthy class. The ten classes are tomato healthy (TH), tomato mosaic virus (TMV), tomato early blight (TEB), tomato late blight (TLB), tomato bacterial spot (TBS), tomato leaf mold (TLM), tomato septoria leaf spot (TSLS), tomato target spot (TTS), tomato yellow leaf curl virus (TYLCV) and tomato two spotted spider mite (TTSSM).

3.2 UPCGAN based data augmentation

The volumetric CNN is considered for designing the proposed UPCGAN architecture. The UPCGAN has a generator module, refiner network, and descriptor. The generator produces the image objects and the refiner network is used for correcting and sharpening the created objects based on the learning from the real samples. A group of diverse compound objects from various types and input noise vectors are augmented by GAN. These objects are an obstacle to automatically attaining the label information. To overcome the aforementioned problem, the conditional GAN is used for handling the generator modes. Accordingly, the generator mode's handling is used for specifying the object label. The encoded class vector is used along with a dimension 10 as an extra input layer for performing the conditioning in the generator and discriminator.

The important reason for utilizing the GAN is that it generates a huge amount of different samples. But, the general issue of the GAN is the mode collapse. In that case, the generator module generates one or small subclasses of anticipated output. To overcome the mode collapse issue, the progressive augmentation of GAN is developed for gradually training the discriminator. The augmented input data with various resolutions are obtained from the generator which are given as input to the discriminator in the progressive training process. The descriptor gradually enhances the classifications based on the studied data dissemination between low and high resolutions samples over the UPCGAN. The feature space augmentation (FSA) approach is used in the discriminator for designing the UPCGAN. In FSA, the augmented synthetic input data are combined with the real sample's feature representation in the intermediate hidden layers of the discriminator. The UPCGAN is used to map the images from one domain to another domain in a paired manner. The proposed UPCGAN has the ability to estimate the uncertainty with the created images. Moreover, UPCGAN performs pixel wise residual distribution using the independent distributed zero mean generalized Gaussian distribution (GGD). The GGD has a tendency to learn and predict the optimal scale and shape of every pixels of the leaf image. The multiphase images are created using UPCGAN with uncertainty estimates. The output obtained from one phase of the image is provided as the input to the next phase of UPCGAN. The inclusion of uncertainty principle in PCGAN helps to refine the regions which are seems to be poorly synthesized and helps in effective classification of diseased leaves.

The 4 convolutional layers of kernel size $4 \times 4 \times 4$ are included in the generator G along with the kernel strides are $\{1, 2, 2, 2\}$ and the number of channels of $\{256, 128, 64, 1\}$. After each layer, the batch normalization and rectified linear unit (ReLU) activation layer are utilized as well as the sigmoid layer $(\sigma(x) = 1/1 + e^{-x})$ is installed in the generator where x is real sample. The discriminator D reflects a generator, however, it utilizes the leaky ReLU instead of conventional ReLU for performing continuous learning, even when there is a dead neuron. A dense layer with N length (N denotes the number of classes) is incorporated for predicting the object class according to the softmax classifier in the classification of object label.

The condition vector y_i and input noise z are integrated into a joint hidden depiction and given as input to the generator. The synthetic objects with the class label and real objects are passed to the discriminator for evaluating and classifying the synthetic object. The objective function for computing the adversarial loss of typical GAN is expressed in Eq. (1).

$$\min_G \max_D V(G, D) = E_{x_i \in p_{data}(x)}[\log D(x_i)] + E_{z \in p_z(z)}[\log(1 - D(G(z)))] \quad (1)$$

Where, the binary cross-entropy function is denoted as $V(G, D)$; expectation operator is denoted as E; the dissemination of the real and the synthetic samples are represented as $p_{data}(x)$ and $p_z(z)$ respectively. Eq. (2) shows the GAN's adversarial loss with a conditional input y_i.

$$\min_G \max_D V(G, D) = E_{x_i \in p_{data}(x)}[\log D(x_i|y_i)] + E_{z \in p_z(z)}[\log(1 - D(G(z|y_i)))] \quad (2)$$

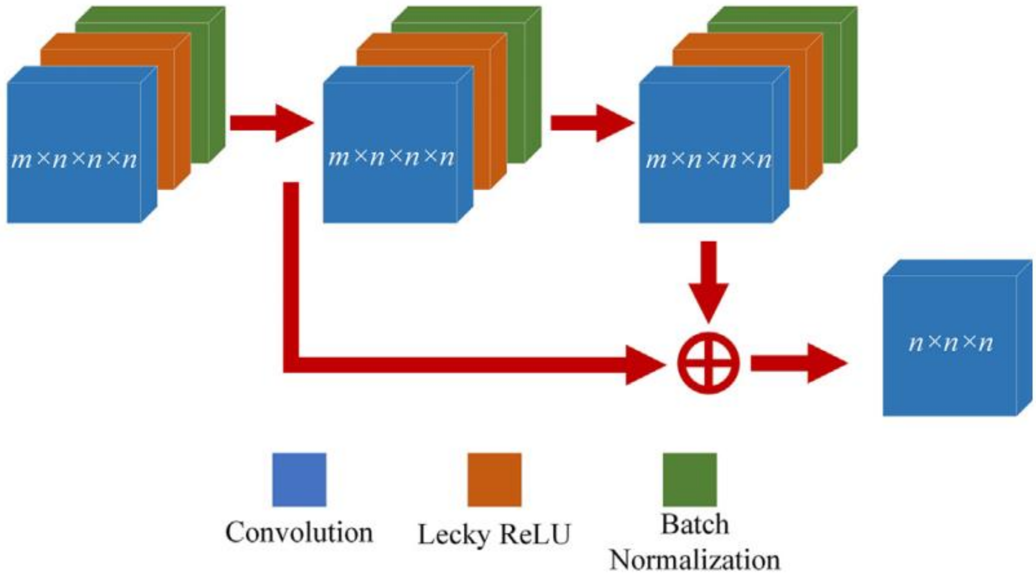

Figure. 2 The architecture of refiner network

The adversarial theory states that the generator makes an effort to cheat the discriminator by classifying their objects as actual by trying to decrease the objective function. However, the discriminator increases the objective function, so there is no chance for misclassification. These are the features of the discriminator and generator from a min-max approach mentioned in Eq. (2). Additionally, the object classification with UPCGAN in the framework of supervised learning is designed and the generator produces the labeled data samples based on the label and noise vector. Hence, the synthetic/fake sample is expressed as $\hat{x}_i = G(z, y_i)$. There are two different probabilities obtained from the discriminator such as probability dissemination over sources i.e., $P(real|x_i)$ and $P(real|\hat{x}_i)$ and probability dissemination over the class label i.e., $P(y_i|x_i)$ and $P(y_i|\hat{x}_i)$. Therefore, the objective function for UPCGAN is expressed as shown in Eqs. (3) and (4).

$$L_D = E[\log P(real|x_i)] + E[\log P(fake|\hat{x}_i)] + E[\log P(y_i|x_i)] + E[\log P(y_i|\hat{x}_i)] \quad (3)$$

$$L_G = E[\log P(real|\hat{x}_i)] + E[\log P(y_i|\hat{x}_i)] \quad (4)$$

In UPCGAN, the L_G is minimized by training the generator and L_D is maximized by training the discriminator. The 1st two terms of Eq. (3) denote classifying both the real and fake samples precisely whereas the last two samples denote that both samples have accurate class labels. For the generator in Eq. (4), all the produced samples are predicted to be categorized as fake using the discriminator as well as it has an accurate class label. Further, the refiner network is added among the generator and the discriminator for obtaining the improved classification. The designed refiner network has 4 blocks from the ResNet and it is illustrated in Fig. 2, where m refers the amount of the filters and n denotes the volume size.

The size of the input volume (n) is similar to the augmented sample's volume size acquired from the generator. Each convolution block used 16 convolutional filters for optimizing the memory. The main objective of the refiner network is used for enhance the realism of fake samples and create them to be identical to the real images. Therefore, the training with augmented samples for enhancing the capacity of feature learning and maximizes accuracy.

3.3 Classification of DenseNet121

After performing the augmentation, the real images and augmented images from UPCGAN are given as input to the DenseNet121 for classifying the tomato leaf diseases. Generally, each layer of DenseNet is linked to each layer in a feed-forward way. This DenseNet121 has four dense blocks which take 224×224 image pixels as input. Here, the 1st convolution layer has 2000 convolution functions with 7×7 size and stride 2. Next, a max pooling layer of 3×3 is used with stride 2. Further, a pooling layer and 3 dense blocks are used where each dense block is placed with the transition layer. The classification layer exists after the fourth dense block. The Fully-connected and softmax layers are eliminated for fine-tuning the pre-trained DenseNet over the images. Next, 2 Convolutional layers with

Table 1. Performance analysis of classifiers without data UPCGAN

Case	Classifier without data augmentation	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Case 1	VGG16	94.08	94.62	95.24	94.68
	VGG19	95.79	96.01	95.33	96.37
	AlexNet	92.36	92.64	93.46	94.29
	DenseNet121	97.11	98.25	97.06	98.32
Case 2	VGG16	93.86	94.11	94.82	93.95
	VGG19	95.47	95.45	96.08	95.91
	AlexNet	91.28	91.34	90.26	91.56
	DenseNet121	96.24	97.08	96.37	97.11
Case 3	VGG16	92.09	92.86	93.01	93.27
	VGG19	93.15	93.22	94.63	93.98
	AlexNet	90.68	91.26	90.08	91.72
	DenseNet121	94.56	96.31	95.94	96.94

ReLU, a fully-connected layer, an average pooling layer, and a softmax layer are incorporated in the DenseNet121. Here, the DenseNet121 is trained to form 100 epochs with a batch size of 32 and a learning rate of 0.0001 where Adam's optimizer is used to update the weights.

4. Results and discussion

The outcomes of the UPCGAN-DenseNet121 method are explained in this section. The UPCGAN-DenseNet121 method is designed and simulated in the Python 3.7 software where the system is operated with 8GB RAM and an i5 processor. The UPCGAN-based data augmentation along with DenseNet121 based classification is proposed for improving the classification of tomato leaf diseases. The performance metrics such as accuracy, precision, recall, and F1-score expressed in Eqs. (5) to (8) are used to evaluate the UPCGAN-DenseNet121.

$$Accuracy = \frac{TP+TN}{TN+TP+FN+FP} \times 100 \quad (5)$$

$$Precision = \frac{TP}{TP+FP} \times 100 \quad (6)$$

$$Recall = \frac{TP}{TP+FN} \times 100 \quad (7)$$

$$F1-score = \frac{2TP}{2TP+FP+FN} \times 100 \quad (8)$$

Where, TP is true positive; TN is true negative; FP is false positive and FN is false negative.

Detailed information about the dataset and performance evaluation are given in the following sections.

4.1 Dataset

The images of the tomato PlantVillage dataset [22] are resized into 224×224 to achieve faster computations. From each class of dataset, 300 images are randomly taken to avoid the data imbalance issue. Therefore a total of $10 \times 300 = 3000$ images are taken for evaluation. At the end of the augmentation performed by UPCGAN, 12,000 images are obtained and these images are combined with real images of the dataset. Hence, a total of 15,000 images are processed by DenseNet121 for classification. The dataset is divided into 60:10:30 as a training set, validation set, and test set for classification. Consider, there is no overlapping is occurred among the three sets. The training and validation sets are used to accomplish the training whereas the testing set is used to evaluate the classification.

4.2 Performance evaluation of UPCGAN-DenseNet121

The performance of UPCGAN-DenseNet121 is evaluated for two different cases where different classes of tomato leaf disease are taken which are specified as follows:

- **Case 1:** In this case, five classes from the dataset such as TH, TLB, TSLS, TTS and TYLCV are taken for analysis.
- **Case 2:** This case 2 considers six different classes such as TH, TSLS, TBS, TLB, TTS and TYLCV for analysis.
- **Case 3:** It considers all 10 classes from the dataset.

For the aforementioned cases, the performances of UPCGAN-DenseNet121 are evaluated with different classifiers such as VGG16, VGG19, and AlexNet. In this section, the performances are evaluated for classifiers without data augmentation and classifiers with UPCGAN. The performance of UPCGAN-DenseNet121 without data augmentation for all three cases is given in Table 1. Next, the graphical illustration of case 3 performances is shown in Fig. 3. From the analysis, it is known that the DenseNet121 without UPCGAN provides higher

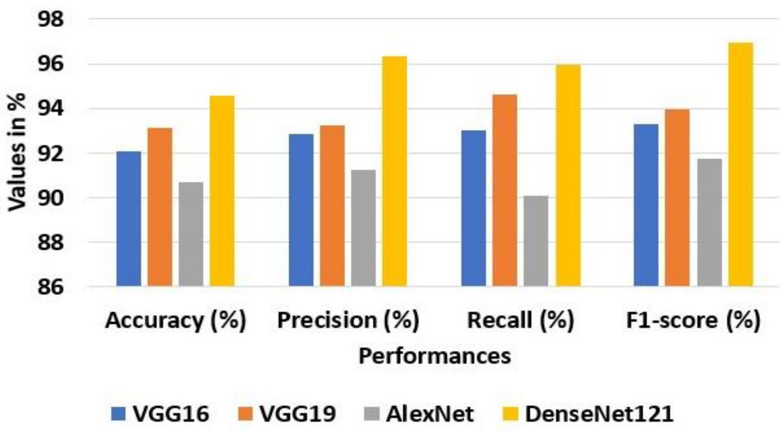

Figure. 3 Graphical results of classifiers without UPCGAN for case 3

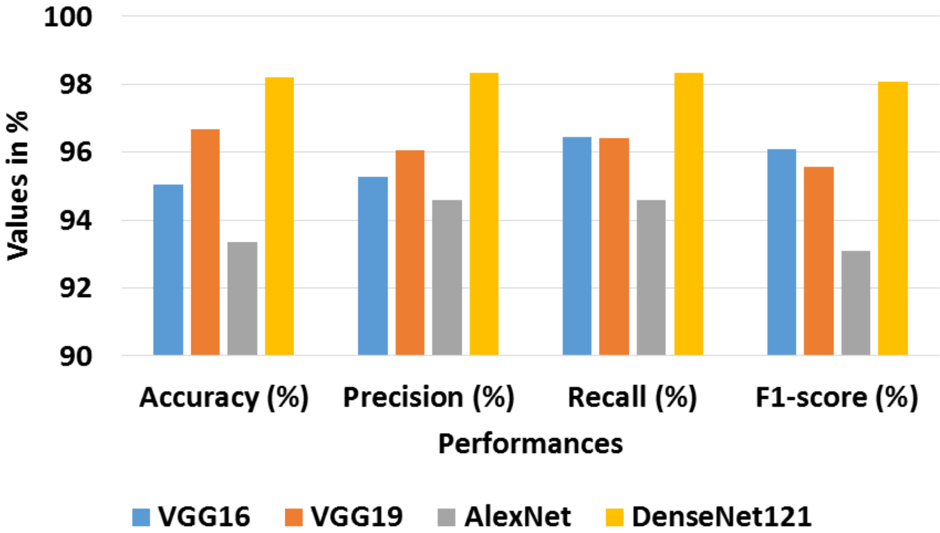

Figure. 4 Graphical results of classifiers with UPCGAN for case 3

classification accuracy than the VGG16, VGG19, and AlexNet for all three cases. For example, the accuracy of DenseNet121 for case 1 is 97.11%, whereas VGG16 obtains 94.08%, VGG19 obtains 95.79% and AlexNet obtains 92.36%. The DenseNet121 achieves better classification because of its higher amount of fully connected layers which helps to achieve a better representation of deeply hidden features.

Table 2 shows the performance analysis of different classifiers (VGG16, VGG19, AlexNet and DenseNet121) with proposed UPCGAN based data augmentation. Next, the graphical results of classifiers with UPCGAN for case 3 are shown in Fig. 4. From the analysis, it is concluded that the combination of UPCGAN and DenseNet121 provides better performance than the other classifiers. For example, the accuracy of UPCGAN-DenseNet121 for case 1 is 99.08%, whereas VGG16 obtains 96.93%, VGG19 obtains 98.82% and AlexNet obtains 95.08%. Moreover, the accuracy of UPCGAN-DenseNet121 is higher than the accuracy of DenseNet121 without UPCGAN for all three cases. The augmented images with different resolutions obtained from the generator of UPCGAN are given as input to the discriminator. Therefore, the progressive training developed in this UPCGAN is used to enhance the classification of tomato leaf diseases.

4.3 Comparative analysis

Existing research such as DCGAN-GoogleNet [17], CGAN-DenseNet121 [20] and FWDGAN [21] are used to evaluate the efficiency of the UPCGAN-DenseNet121 method for all three cases mentioned in the previous section. The DCGAN-GoogleNet [17] and FWDGAN [21] are implemented for the same cases mentioned in section 4.2 for comparison. The comparative analysis of UPCGAN-DenseNet121 method is shown in the Table 3. Additionally, the graphical comparison of F1-score for UPCGAN is shown in Fig. 5. From the comparisons, it is

Received: May 12, 2023. Revised: June 13, 2023.

Table 2. Performance analysis of classifiers with UPCGAN

Case	Classifier with UPCGAN	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Case 1	VGG16	96.93	95.64	96.93	96.44
	VGG19	98.82	97.22	97.06	96.79
	AlexNet	95.08	95.93	94.28	94.77
	DenseNet121	99.08	98.74	99.21	99.03
Case 2	VGG16	96.05	95.08	96.7	96.17
	VGG19	97.84	97.08	97.46	96.63
	AlexNet	94.47	94.84	95.05	95.08
	DenseNet121	98.64	98.5	98.74	98.93
Case 3	VGG16	95.05	95.28	96.46	96.1
	VGG19	96.68	96.05	96.43	95.58
	AlexNet	93.35	94.58	94.58	93.09
	DenseNet121	98.2	98.34	98.34	98.06

Table 3. Comparative analysis of UPCGAN-DenseNet121

Case	Methods	Case		
		Case 1	Case 2	Case 3
Accuracy (%)	DCGAN-GoogleNet [17]	94.33	92.04	90.48
	CGAN-DenseNet121 [20]	99.51	98.65	97.11
	FWDGAN [21]	98.71	96.48	95.14
	UPCGAN-DenseNet121	99.08	98.64	98.2
Precision (%)	DCGAN-GoogleNet [17]	93.47	91.86	90.35
	CGAN-DenseNet121 [20]	99	98	97
	FWDGAN [21]	98.09	96.27	96.83
	UPCGAN-DenseNet121	98.74	98.5	98.34
Recall (%)	DCGAN-GoogleNet [17]	94.70	92.09	89.34
	CGAN-DenseNet121 [20]	99	99	97
	FWDGAN [21]	97.19	95.47	96.97
	UPCGAN-DenseNet121	99.21	98.74	98.34
F1-score (%)	DCGAN-GoogleNet [17]	94.08	92.43	89.97
	CGAN-DenseNet121 [20]	99	98	97
	FWDGAN [21]	97.72	95.52	95.53
	UPCGAN-DenseNet121	99.03	98.93	98.06

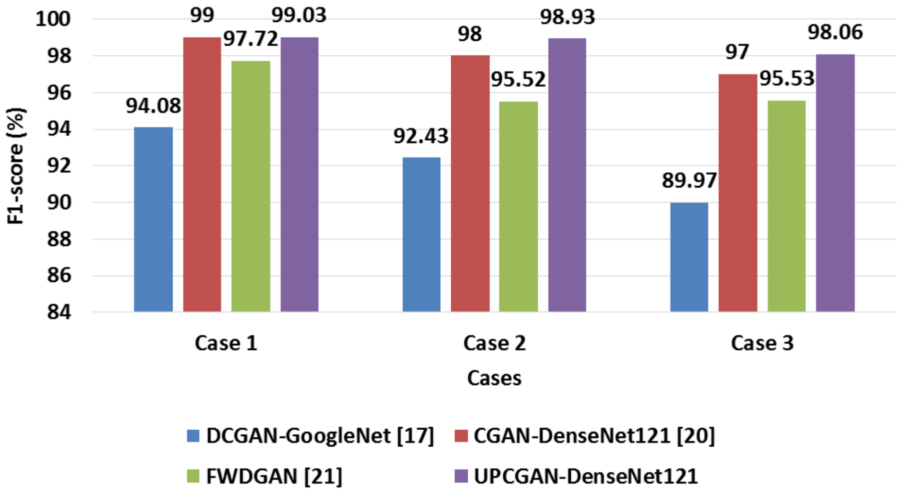

Figure. 5 Graphical comparison of F1-score for UPCGAN

concluded that the UPCGAN-DenseNet121 outperforms well than the DCGAN-GoogleNet [17], CGAN-DenseNet121 [20] and FWDGAN [21]. The performances of UPCGAN-DenseNet121 slightly varied when compared to the CGAN-DenseNet121 [20], but the CGAN-DenseNet121 [20] provides lesser performances when it analysed with huge amount of data samples. Specifically, the higher F1-score of the UPCGAN-DenseNet121 shows that it does not consider the data imbalance during the classification. For instance, the UPCGAN-DenseNet121 has obtained 98.06% F1-score for third case whereas DCGAN-GoogleNet [17] obtains as 89.97%, CGAN-DenseNet121 [20] obtains as 97% and FWDGAN [21] obtains as 95.53 %. The better result of the proposed UPCGAN-DenseNet121 is due to the pixel wise residual distribution which predict the optimal scale and shape of every pixel of the leaf image and aids in better classification.

5. Conclusion

Many applications for the computerized diagnosis of plant leaf disease are developed according to deep learning techniques. However, the existing applications are suffered from overfitting issues, because of insufficient training data. In this paper, UPCGAN based augmentation is developed for generating synthetic images (augmented images) based on the input samples. The progressive learning of UPCGAN utilizes the lower to higher resolutions of input images for gradually improving the generator performance and generating informative objects. Further, the original images are trained along with the synthetic images for improving the classification of plant leaf diseases using DenseNet121. The augmented training samples with various resolutions are used to enhance the classification performances. From the results, it is concluded that the UPCGAN-DenseNet121 provides better performance than the DCGAN-GoogleNet, CGAN-DenseNet121 and FWDGAN. The accuracy of UPCGAN-DenseNet121 for 10 classes is 98.2%, which is high when compared to the DCGAN-GoogleNet, CGAN-DenseNet121 and FWDGAN. In future, the deep learning with optimal learning rate for improving the classification of leaf diseases.

Notation

Parameter	Description
G	Generator
σ	Sigmoid layer
x	Real sample
N	Length of dense layer
y_i	Condition vector
z	Input noise
$V(G,D)$	Binary cross-entropy function
E	Expectation operator
$p_{data}(x)$	Dissemination of the real samples
$p_z(z)$	Dissemination of the synthetic samples
$\hat{x}_i$	Synthetic/fake sample
$P(.)$	Probability dissemination
L	Objective function for UPCGAN
n	Size of the input volume
TP	True positive
TN	True negative
FP	False positive
FN	False negative

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

For this research work all authors' have equally contributed in Conceptualization, methodology, validation, resources, writing—original draft preparation, writing—review and editing.

References

[1] M. M. Monowar, M. A. Hamid, F. A. Kateb, A. Q. Ohi, and M. F. Mridha, “Self-Supervised Clustering for Leaf Disease Identification”, *Agriculture*, Vol. 12, No. 6, p. 814, 2022.

[2] H. Amin, A. Darwish, A. E. Hassanien, and M. Soliman, “End-to-End Deep Learning Model for Corn Leaf Disease Classification”, *IEEE Access*, Vol. 10, pp. 31103-31115, 2022.

[3] Ü. Atila, M. Uçar, K. Akyol, and E. Uçar, “Plant leaf disease classification using EfficientNet deep learning model”, *Ecological Informatics*, Vol. 61, p. 101182, 2021.

[4] Q. H. Cap, H. Uga, S. Kagiwada, and H. Iyatomi, “LeafGAN: An Effective Data Augmentation Method for Practical Plant Disease Diagnosis”, *IEEE Transactions on Automation Science and Engineering*, Vol. 19, No. 2, pp. 1258-1267, 2022.

[5] 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.

[6] Q. Zeng, X. Ma, B. Cheng, E. Zhou, and W. Pang, “GANs-Based Data Augmentation for Citrus Disease Severity Detection Using Deep Learning”, *IEEE Access*, Vol. 8, pp. 172882-172891, 2020.

[7] P. Kaur, S. Harnal, V. Gautam, M. P. Singh, and S. P. Singh, “A novel transfer deep learning method for detection and classification of plant leaf disease”, *Journal of Ambient Intelligence and Humanized Computing*, 2022.

[8] S. Ahmed, M. B. Hasan, T. Ahmed, M. R. K. Sony, and M. H. Kabir, “Less is More: Lighter

and Faster Deep Neural Architecture for Tomato Leaf Disease Classification", *IEEE Access*, Vol. 10, pp. 68868-68884, 2022.

[9] S. Nandhini and K. Ashokkumar, "An automatic plant leaf disease identification using DenseNet-121 architecture with a mutation-based henry gas solubility optimization algorithm", *Neural Computing and Applications*, Vol. 34, No. 7, pp. 5513-5534, 2022.

[10] A. Yadav, U. Thakur, R. Saxena, V. Pal, V. Bhateja, and J. C. W. Lin, "AFD-Net: Apple Foliar Disease multi classification using deep learning on plant pathology dataset", *Plant and Soil*, Vol. 477, Nos. 1-2, pp. 595-611, 2022.

[11] D. K. Mahato, A. Pundir, and G. J. Saxena, "An Improved Deep Convolutional Neural Network for Image-Based Apple Plant Leaf Disease Detection and Identification", *Journal of The Institution of Engineers (India): Series A*, Vol. 103, No. 4, pp. 975-987, 2022.

[12] D. Su, H. Kong, Y. Qiao, and S. Sukkarieh, "Data augmentation for deep learning based semantic segmentation and crop-weed classification in agricultural robotics", *Computers and Electronics in Agriculture*, Vol. 190, p. 106418, 2021.

[13] M. Nagaraju, P. Chawla, S. Upadhyay, and R. Tiwari, "Convolution network model based leaf disease detection using augmentation techniques", *Expert Systems*, Vol. 39, No. 4, p. e12885, 2022.

[14] B. Liu, C. Tan, S. Li, J. He, and H. Wang, "A Data Augmentation Method Based on Generative Adversarial Networks for Grape Leaf Disease Identification", *IEEE Access*, Vol. 8, pp. 102188-102198, 2020.

[15] H. Nazki, S. Yoon, A. Fuentes, and D. S. Park, "Unsupervised image translation using adversarial networks for improved plant disease recognition", *Computers and Electronics in Agriculture*, Vol. 168, p. 105117, 2020.

[16] J. A. Pandian, V. D. Kumar, O. Geman, M. Hnatiuc, M. Arif, and K. Kanchanadevi, "Plant Disease Detection Using Deep Convolutional Neural Network", *Applied Sciences*, Vol. 12, No. 14, p. 6982, 2022.

[17] Q. Wu, Y. Chen, and J. Meng, "DCGAN-Based Data Augmentation for Tomato Leaf Disease Identification", *IEEE Access*, Vol. 8, pp. 98716-98728, 2020.

[18] C. Zhou, Z. Zhang, S. Zhou, J. Xing, Q. Wu, and J. Song, "Grape leaf spot identification under limited samples by fine grained-GAN", *IEEE Access*, Vol. 9, pp. 100480-100489, 2021.

[19] H. Deng, D. Luo, Z. Chang, H. Li, and X. Yang, "RAHC_GAN: A Data Augmentation Method for Tomato Leaf Disease Recognition", *Symmetry*, Vol. 13, No. 9, p. 1597, 2021.

[20] A. Abbas, S. Jain, M. Gour, and S. Vankudothu, "Tomato plant disease detection using transfer learning with C-GAN synthetic images", *Computers and Electronics in Agriculture*, Vol. 187, p. 106279, 2021

[21] M. Li, G. Zhou, A. Chen, J. Yi, C. Lu, M. He, and Y. Hu, "FWDGAN-based data augmentation for tomato leaf disease identification", *Computers and Electronics in Agriculture*, Vol. 194, p. 106779, 2022.

[22] D. Hughes and M. Salathé, "An open access repository of images on plant health to enable the development of mobile disease diagnostics", arXiv preprint arXiv:1511.08060.