Detection of rice plant diseases based on deep transfer learning

4 authors, including:

Junde Chen
Xiamen University
67 PUBLICATIONS **2,314** CITATIONS

SEE PROFILE

Defu Zhang
Xiamen University
138 PUBLICATIONS **5,212** CITATIONS

SEE PROFILE

Yaser Ahangari Nanehkaran
School of Information Engineering
61 PUBLICATIONS **1,741** CITATIONS

SEE PROFILE

SCI

Detection of rice plant diseases based on deep transfer learning

Junde Chen,[a] ● Defu Zhang,[a*] Yaser A Nanehkaran[a] and Dele Li[b]

Abstract

BACKGROUND: As the primary food for nearly half of the world's population, rice is cultivated almost all over the world, especially in Asian countries. However, the farmers and planting experts have been facing many persistent agricultural challenges for centuries, such as different diseases of rice. The severe rice diseases may lead to no harvest of grains; therefore, a fast, automatic, less expensive and accurate method to detect rice diseases is highly desired in the field of agricultural information.

RESULTS: In this article, we study the deep learning approach for solving the task since it has shown outstanding performance in image processing and classification problem. Combining the advantages of both, the DenseNet pre-trained on ImageNet and Inception module were selected to be used in the network, and this approach presents a superior performance with respect to other state-of-the-art methods. It achieves an average predicting accuracy of no less than 94.07% in the public dataset. Even when multiple diseases were considered, the average accuracy reaches 98.63% for the class prediction of rice disease images.

CONCLUSIONS: The experimental results prove the validity of the proposed approach, and it is accomplished efficiently for rice disease detection.
© 2020 Society of Chemical Industry

Keywords: rice disease detection; convolutional neural networks; transfer learning; image classification

INTRODUCTION

As a global scale, more than half the world's population regards rice as the primary staple. Rice diseases have significant negative effects on the quality and quantity of rice production.[1] Also, they are major threats to food security. Obtaining information about real-time rice growth and health is highlighted in modern agriculture. Based on the information of rice diseases obtained from different sources, rice diseases can be forecasted before their outbreak during the process of rice production.[2] The conventional identification of rice diseases is mainly based on visual observations of experienced producers or rice experts in the field. This requires continuous monitoring of experts which might be prohibitively expensive in large farms.[3,4] Especially, in some developing countries, farmers may have to travel long distances to contact agricultural experts, which is undoubtedly time-consuming, costly, lags in prediction, and cannot be carried out in a wide range. Nevertheless, with the rapid development of image processing and pattern recognition technologies, the new way for plant disease recognition and diagnosis is provided. The image recognition and machine learning are the essential research topics as they may prove beneficial in monitoring large fields of crops, and thus automatically detect the symptoms of diseases as soon as they appear on plant leaves.[5–7]

Plenty of previous works have considered the image recognition, and a particular classifier is used which discriminates the image categories. In the past decades, automatic image recognition technology has shown powerful performance along with the improvement of digital camera and the increasing computational capacity, which have been applied in many fields including

medical image processing,[8,9] food analysis,[10,11] biometrics,[12] industrial production,[13,14] emotion learning,[15] among others[16–18] and have achieved excellent results. Generally, the plant leaves are the first source to detect most plant diseases, which can be detected automatically through efficient image recognition techniques.[19,20] Until now, some research works have been also involved in this field. Amit Kumar Singh et al.[21] used support vector machine (SVM) which classified normal and diseased rice leaf into two different classes, and their classifier achieved an accuracy of more than 82%. Phadikar et al.[22] used two approaches to classify rice diseases, including the Bayes and SVM classifiers. In their work, ten different combinations of training and test data resulted in an accuracy of 79.5%. In fact, the SVM achieved 68.1% accuracy with ten-fold cross-validation. Kahar et al.[23] used the artificial neural network (ANN) as a classifier to perform the disease detection of rice, in which three output nodes were used for detecting three diseases: leaf blight, leaf blast, and sheath blight. Their identification scheme achieved an accuracy of 74.21%. Orillo et al.[24] used 30 rice leaf images for each disease, their system used the backpropagation neural network for classification, etc. More recently, deep learning techniques, particularly convolutional neural networks (CNNs), are quickly becoming the preferred method to

* Correspondence to: D Zhang, School of Informatics, Xiamen University, Xiamen 361005, China. E-mail: dfzhang@xmu.edu.cn

a School of Informatics, Xiamen University, Xiamen, China

b Fujian College of Water Conservancy and Electric Power, Sanming, China

overcome some challenges.[25] So far, there has been little attention to using a deep learning approach in agriculture.[26] Most of the existing studies focused on disease recognition from a public dataset of images zoomed on crop leaves.[27–29] For example, Mohanty et al.[30] trained a deep learning model for recognizing 14 crop species and 26 crop diseases. Their trained model achieves an accuracy of 99.35% on a held-out test set. Shrivastava et al.[31] used the transfer learning of deep CNN to conduct the detection of three common rice plant diseases: rice blast, bacterial leaf blight, and sheath blight. They realized the classification accuracy of 91.37%. Ma et al.[32] used a deep CNN to conduct symptom-wise recognition of four cucumber diseases, such as, anthracnose, downy mildew, powdery mildew, and target leaf spots. They reached the recognition accuracy of 93.4%. Applying the transfer learning of the Inception-V3 neural network, Qiang et al.[33] achieved the precision rate of 93% on the test set for the crop disease detection, and they proved the effectiveness of Inception-V3 based deep learning model. Kawasaki et al.[34] introduced a system based on CNN to recognize cucumber leaf disease; it reaches an accuracy of 94.9%, etc. Although very good results have been reported in the literature, investigations so far have used image databases with limited diversity. The most photographic materials include solely images in experimental (laboratory) setups, not in a real field scenario. In fact, images are captured under a wide diversity of conditions and include an extensive variety of symptom characteristics.[25] Despite the limitations, the previous investigations have successfully demonstrated the potential of deep learning algorithms in plant disease detection.

In this work, we studied the transfer learning for the deep CNNs and modified the model with the aim of enhancing the learning ability of the tiny lesion features without increasing the computational complexity. Through combining the Inception module[35] and the pre-trained DenseNet[36] on ImageNet,[37] we performed the transfer learning for the target dataset. The top layers of DenseNet were truncated by defining a new Inception module with a fully-connected ReLU layer of 512 neutrons, then, a fully-connected Softmax layer with the practical number of categories was added as the top layer of the modified network, more than that, instead of the original Cross-Entropy Loss function, a boost Focal Loss function was utilized in the network to enhance the learning ability of the lesion features. Thus, the new network structure, which we term the DENS-INCEP, was formed and used for the class prediction of rice disease images.

The remainder of this article is organized in the following manner. The next section presents the collection of the image dataset followed by an overall flow introduction, and this section mainly discusses the methodology to accomplish the task of rice disease detection along with related concepts and the proposed approach. The third section is dedicated to the algorithm experiments, multiple experiments are conducted and the experimental results are evaluated as well as comparative analysis. Finally, the article is summarized in the fourth section.
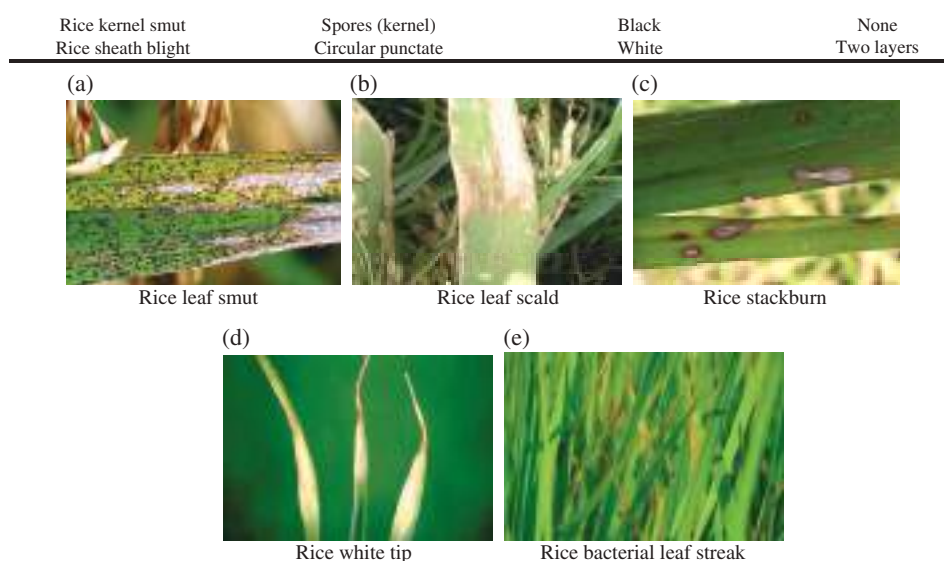
MATERIALS AND METHODS

Data acquisition
About 500 rice plant disease images were captured from the experimental field of agricultural scientific innovation base, Fujian Institute of Subtropical Botany, Xiamen, China. All the images have been labeled in categories and are saved as JPG format. The background conditions of images are complex, and the illumination intensities are uneven. For the subsequent calculations, these images are uniformly processed into the RGB model by Photoshop tools firstly, and then the size of images is adjusted. These rice disease images mainly include the rice stackburn, rice leaf scald, rice leaf smut, rice white tip, and rice bacterial leaf streak, etc. Some of the sample images are displayed in Fig. 1(a–e) and the symptom characteristics of various rice leaf diseases are given in Table 1.

Overview
The overall process of our method for rice disease detection is presented as follows: initially, the samples of rice disease images are collected and labeled based on the experts' knowledge in the field; then, the image-processing techniques including image resizing, image sharpening and image edge filling, etc., are performed on the acquired images, and new sample images are generated to enrich the dataset using the data augmentation methods, for example, rotation and translation are utilized to generate the augmented dataset; after that, the sample images are

Rice kernel smut	Spores (kernel)	Black	None
Rice sheath blight	Circular punctate	White	Two layers
(a)	(b)	(c)	

Rice leaf smut   Rice leaf scald   Rice stackburn

(d)   (e)

Rice white tip   Rice bacterial leaf streak

Figure 1. Sample images of rice leaf diseases. (a) Rice leaf smut, (b) rice leaf scald, (c) rice stackburn, (d) rice white tip, and (e) rice bacterial leaf streak.

Table 1. The symptom characteristics of rice disease images

Disease name	The leaf disease characteristics		
	Shape	Color	Junction
Rice stackburn	Circular or Ellipse	Yellowish-white macular	None
Leaf scald	Watery stain form or strip	Off-white or yellow	None
Leaf smut	Short strip	Black	None
White tip	Strip	Centre: ashen; edge: brown	Two layers
Bacterial leaf streak	Long strip	Brown or russet	Two layers
Rice false smut	Spores (grain)	Black	None
Rice blast	Circular punctate	Brown	Two layers
Rice stem rot	Long strip (stem)	Black	None
Rice sheath spot	Strip (sheath)	Centre: off-white to brown	Two layers
Rice sheath rot	Long strip(sheath)	Brown	None
Grain spotting and peck	Circular (grain)	Centre: off-white to brown	Two layers
Rice kernel smut	Spores (kernel)	Black	None
Rice sheath blight	Circular punctate	White	Two layers

input to the proposed method for model training, and then the trained model is applied for the class prediction of rice disease images. Thus, the final detection results are obtained and can be used to update the sample knowledge library, as depicted in Fig. 2. The detailed descriptions of these phases are illustrated in subsequent sections.

Related works
DenseNet
DenseNet is the deep CNN architecture published by Huang *et al.*[36] in the proceedings of 'Computer Vision and Pattern Recognition' (CVPR) 2017. To alleviate the vanishing-gradient problem with the increase of network depth, DenseNet connects the output of all the layers to the input of all the following layers within the dense block. A dense block is the repetition of a Batch Normalization, a ReLU, a 1×1 convolution, a Batch Normalization, a ReLU, and a 3×3 convolution a specific number of times, as we can see a three-layer dense block in Fig. 3. After each dense block, the Translation Layer reduces the action of the dimension as a 1×1 convolution pooling layer.

For each layer, the feature-maps of all preceding layers are used as inputs and its own feature-maps are used as inputs into all subsequent layers, as expressed in Eqn (1).

$$x_l = H_l([x_0, x_1, \cdots, x_{l-1}]) \qquad (1)$$

where l denotes the layer number of the CNN, $[x_0, x_1, \ldots, x_{l-1}]$ represents cascading the feature maps from 0 to $l - 1$ layers as the input of the subsequent layer, similar to the operation of the Inception module.[35]

In this way, DenseNet ensures the maximum information flow between layers in the network and avoids the problem of information loss. Mainly, because each layer of the convolution layer in the dense block is small, only 1×1 and 3×3, each convolution kernel is responsible for learning the tiny features. Therefore, in theory, it is suitable for the detection of plant leaf lesions.

Inception module
To improve the capability of deep neural networks, the most straightforward way is to increase the depth or width of the networks. However, with the increase of network depth and width, it contains too many parameters and causes more consumption of computing resources. Therefore, to overcome these problems, the Inception module was first introduced in the GoogLeNet architecture by Szegedy *et al.*[35] and it achieved impressive, record-breaking performance as the winner of the ImageNet ILSVRC challenge competition. The inception module is composed of a max-pooling layer and convolutional layers with the varied sizes of 1×1, 3×3 and 5×5, which is stacked together. Between two parallel 1×1 convolution layer, a max-pooling layer is used to conduct the dimension reduction, and a concatenation filter is required to merge the different layers. Moreover, a variant of inception module, which is changed by removing the 5×5 convolution and introducing instead two 3×3 convolutions, is widely applied in later network architectures, such as Inception v3 and so on. This is depicted in Fig. 4.

Transfer learning
Transfer learning is a kind of deep learning method where a CNN trained for a task is reused as the starting point for a model on a second task.[38] This is because deep learning architectures need huge time and computing resources to train massive parameters, while the collecting of a large labeled dataset for model training is undoubtedly a challenging task. Thus, the scheme of transfer learning has increasingly become the preferred method and is naturally employed in the practical application, that is, the solutions consisting in using a pre-trained network where only the parameters of the last classification layers need to be inferred from scratch using the training set.[39] The main processes of the transfer learning approach are described as follows.

(1) Determine the base network
    The base network needs to be first determined and the network weights (W_1, W_2, …, W_n) are assigned using the pre-trained CNN model, which is trained on a large labeled dataset, such as ImageNet.[37]
(2) Establish a new neural network
    By replacing the layers, inserting the layers and deleting the layers from the bottom layer network, the network can be modified to generate a new network architecture, which can be used to achieve the specific task or target.
(3) Model training or fine-tuning
   Deep neural networks are generally trained based on the prediction loss minimization. With the target dataset X and corresponding labels Y, the objective of model training is to iteratively minimize the average loss E defined in Eqn (2).[40]

$$E(W) = \frac{1}{N} \sum_{i=1}^{N} L(f(W; x_i), y_i) + \lambda R(W) \qquad (2)$$

where N represents the number of data instances (mini-batch) in every iteration, W denotes the weighting matrices of neural
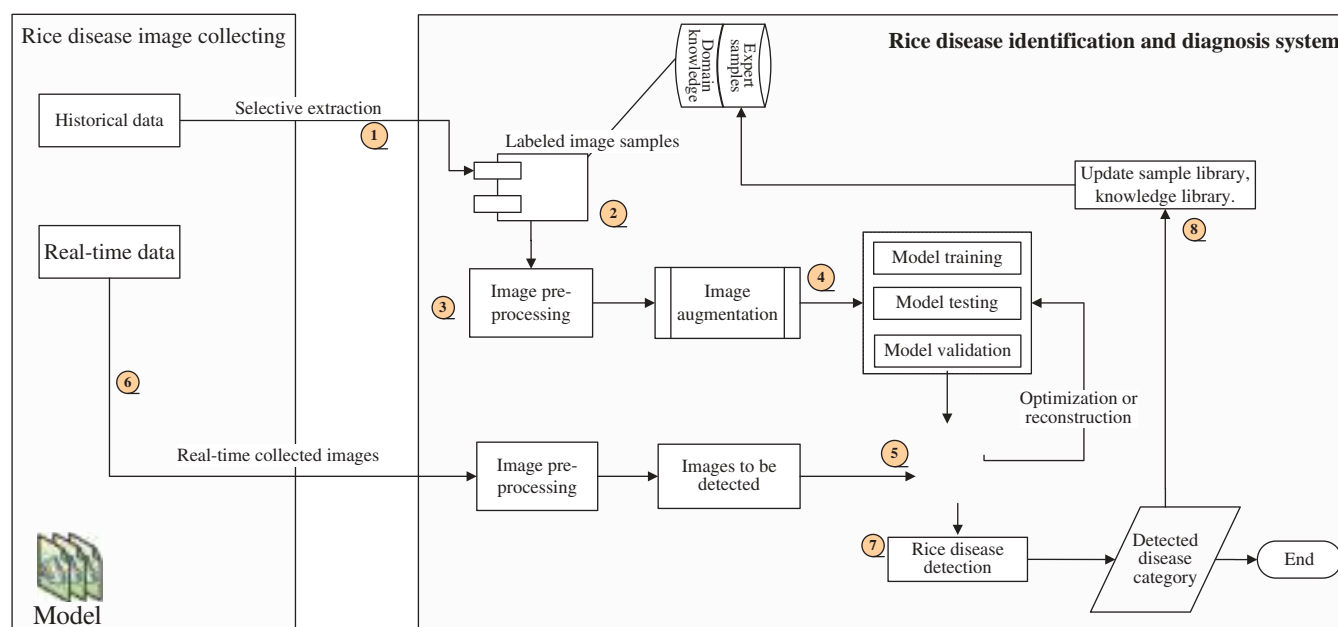
Figure 2. The overall flow of rice disease recognition.
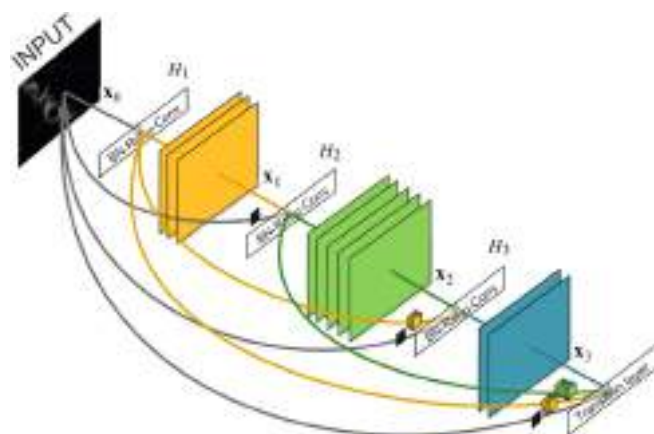
Figure 3. A three-layer dense block with each layer taking all preceding feature-maps as input.

Figure 4. Inception module.

network, $L(\cdot)$ is the loss function, $f(\cdot)$ is the predicted output of neural network, and $R(\cdot)$ is the regularization term or weight decay with Lagrange multiplier λ.

Generally, the stochastic gradient descent (SGD)[40] algorithm is often used to calculate the optimal W by minimizing the loss function E on the target dataset, as defined in Eqn (3).
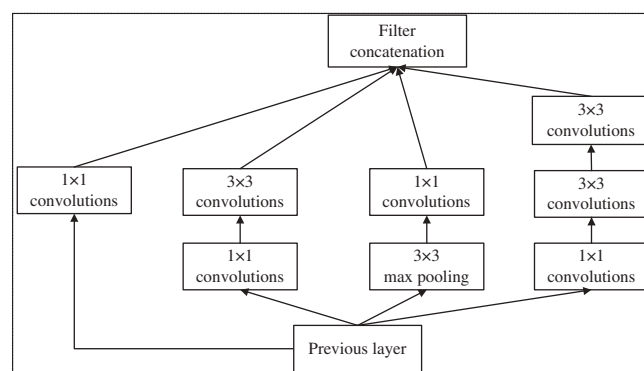
$$W_k = W_{k-1} - a(\partial E(W)/\partial W) \tag{3}$$

where a is the learning rate, and k is the index of classes.

Instead of starting the training from scratch by randomly initializing the weights, the transfer learning can initialize the weights using the pre-trained network or apply the fine-tuning approach. Figure 5 depicts a specific process of transfer learning.

DENS-INCEP network model

As stated earlier, DenseNet represents the current state-of-the-art with outstanding feature extraction ability since the latter feature

layer of this network will receive the features for all the feature layers in front of it, thus reducing feature loss.[36] DenseNet is widely used in practical application and often employed for transfer learning. Moreover, in ILSVRC 2014, the GoogLeNet with Inception module has shown its benefits, which achieves the impressive performance in the competition. Therefore, the pre-trained DenseNet with the Inception module is selected in our transfer learning.

By combining the advantages of both, the pre-trained DenseNet and Inception module were fused to generate a new network, aliased as DENS-INCEP, which was used for the class prediction of plant disease images. We modified the conventional DenseNet by replacing its full connection layer with a global pooling layer, then an Inception module and a new fully-connected ReLU layer with 512 neutrons are introduced in the network, after that, a fully-connected Softmax layer with the practical number of categories was used as the top layer of the modified network, thus the proposed approach mainly comprised of two parts: the first part is the pre-trained module, which is used as a basic feature extractor; the latter is the extended structure that focuses on the multi-scale
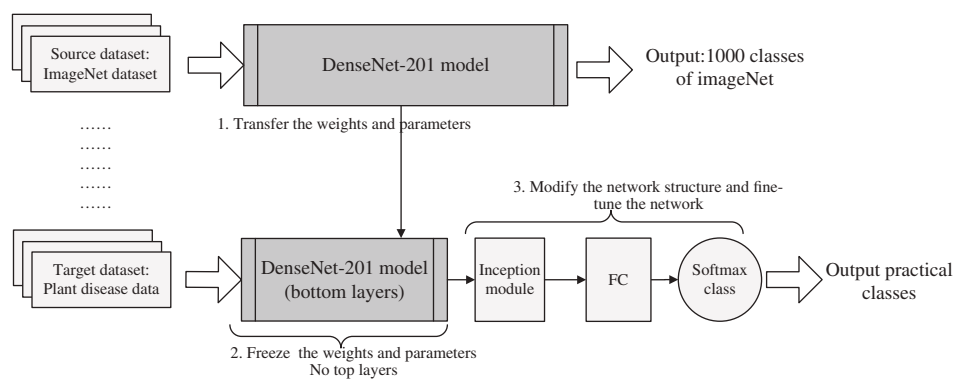
Figure 5. The process of transfer learning.
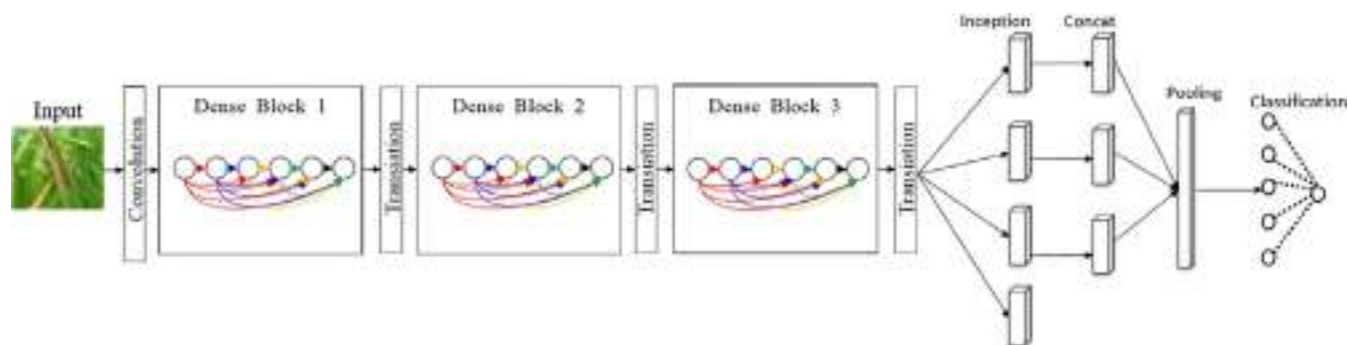
Figure 6. The proposed DENS-INCEP architecture.

feature extraction. Specifically, the network is obtained by applying the following steps.

(1) The first step initializes network parameters using a pre-trained network. DenseNet-201 is pre-trained from the large labeled ImageNet dataset, where it provides good performance in image recognition and classification.
(2) The second step adapts to a new classification problem. To solve the new classification problem, an adaptation is performed by freezing all weights of the pre-trained layers, and the last fully connected layers of the original network are truncated since it was used to classify the 1000 classes of ImageNet.
(3) An extension is made on the model. The original model is extended by adding an Inception module and a new fully-connected ReLU layer with 512 neutrons, thus all the layers of the new model are trained on the new images (target dataset) and the final layer uses the Softmax function for the class prediction of rice disease images. Figure 6 depicts the network structure and relevant parameters are presented in Table 2.

In addition, the Cross-Entropy Loss function is often used as the loss function of the traditional conventional neural networks, and it is calculated in Eqn (4).

$$L = -\sum_{c=1}^{N} y_c \log(p_c) \tag{4}$$

where N represents the number of classifications; y_c is the indicator variable (0 or 1), if the category is the same as the category of

the sample, it is 1, otherwise it is 0; p_c is the predicted probability that the observed sample belongs to category c. Based on this, Lin *et al.*[41] reported the Focal Loss function because the predicted loss weights of the Cross-Entropy function is considered the positive and negative samples as the same. The Focal Loss function is calculated by

$$L(p_c) = -\alpha_c(1-p_c)^{\gamma} \log(p_c) \tag{5}$$

where a_c denotes the weighting factor of Focal Loss function when the corresponding category is 1, and γ is the modulating factor (hyperparameter). The Focal Loss function is useful for the binary classification problem, while the detection of rice disease images is a multi-classification problem. Therefore, on the basis of this, we developed it as a loss function for multi-classification tasks, as calculated in Eqn (6).

$$L_{mult} = -\sum_{i=1}^{C} \alpha_i(1-p_i)^{\gamma} y_i \log(p_i) \tag{6}$$

$$y_i = \begin{cases} 1 & i = acutal\_class \\ 0 & i \neq acutal\_class \end{cases} \tag{7}$$

where C represents the total number of categories, p_i is the predicted probability distribution, and a_i is the weighting factor.

EXPERIMENTAL RESULTS AND ANALYSIS

In our experiments, some image pre-processing algorithms were conducted using Matlab, while the data augmentation and CNN

Table 2. The main parameters of the DENS-INCEP network

Module (type)	Function	Output shape	Kernel size	Stride
Input	None	(None, 224, 224, 3)	None	None
Dense Block1	Convolutions	(None, 224, 224, 64)	$1 \times 1, 3 \times 3$	1
Trans Layer1	Convolution, Pooling	(None, 112, 112, 64)	$1 \times 1, 2 \times 2$	2
Dense Block2	Convolutions	(None, 112, 112, 96)	$1 \times 1, 3 \times 3$	1
Trans Layer2	Convolution, Pooling	(None, 56, 56, 96)	$1 \times 1, 2 \times 2$	2
Dense Block3	Convolutions	(None, 56, 56, 128)	$1 \times 1, 3 \times 3$	1
Trans Layer3	Convolution, Pooling	(None, 28, 28, 128)	$1 \times 1, 2 \times 2$	2
Inception module	Inception module	(None, 28, 28, 512)	$1 \times 1, 3 \times 3$	None
Fully-connect layer	Fully-connect	(None, 512)	None	None
Output	Softmax regression	(None, classes)	Classifier	None
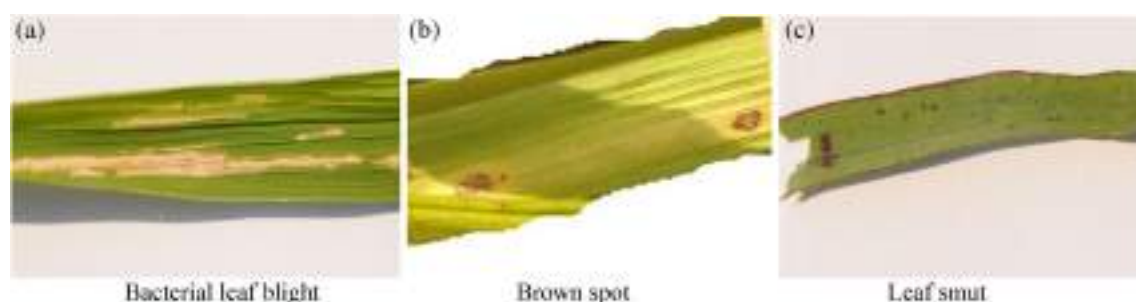
Figure 7. Examples of rice leaf diseases dataset. (a) Bacterial leaf blight, (b) brown spot, and (c) leaf smut.

were implemented using Anaconda3 (Python 3.6), Keras-GPU library,[42] and OpenCV-python3 library. The deep CNN training and testing are accelerated by GPU, and the experimental hardware environment includes: Intel® Xeon(R) E5-2620 v4 central processing unit (CPU) at 2.10 GHz with 64-GB memory and NVIDIA GeForce RTX 2080 (CUDA 10.2) graphics card,[43] which is used for the model training and testing.

Experiments on a public dataset

The UCI repository[44] is an international general database for the algorithm test of machine learning, and the Rice Leaf Diseases dataset is downloaded from the UCI database. This dataset contains 120 color leaf images and divided into three disease classes: Bacterial leaf blight, Brown spot, and Leaf smut separately, with 40 images in each category.[45] It is noted that the images are taken with a simple background, and some images of the same leaf are captured from different orientations. The dimensions of the images are different, and some partial examples of rice disease images are displayed in Fig. 7. Due to the non-massive data of this dataset, new samples were generated to enrich the dataset using data augmentation techniques.[46] The original database is expanded using Python script by random horizontal or vertical flipping, random angle rotation, and random scaling of the original images. In the process of generating new sample images, the rotation, flipping, and scale transform are performed by utilizing some random bounded values. These values are evenly distributed in a specific range, the rotation range is ±15°, the scale is changed from 0.9 to 1.1, and the shear range is 0.2, etc. In this way, except a certain number of raw images are retained to validate the effectiveness of the model, the original images are augmented to at least 200 images per category for the modeling, as we can see in Fig. 8.

Based on the method proposed earlier, we perform the model training and validation on the Rice Leaf Diseases dataset. All the sample images are uniformly resized to the fixed-dimension of 224×224 pixels to fit the model, and the training set and test set are divided according to the ratio of 7:3. Particularly, to further verify the effectiveness of the proposed approach, we considered four influential CNNs including DenseNet,[36] VGGNet,[47] Inception V3[33,35] and ResNet [48] for the comparative experiments. Using the transfer learning, these models are created and loaded with pre-trained weights from ImageNet, and the top layers are truncated by defining a new fully-connected Softmax layer with the practical number of classifications. Additionally, referring to the approach proposed by Shrivastava et al.[31] for transfer learning, the deep CNN was used for extraction of features while SVM for classification, thus the fused VGGNet-19 and SVM (aliased as VGG19-SVM) was also selected for the comparative experiments especially.

Considering the statistics of correct detections (also known as true positives), misdetections (also known as false negatives), true negatives and false positives, we can evaluate the performance of the models with the indicators including the accuracy, sensitivity, and specificity, as expressed in Eqns (8–10).

$$\text{Accuracy} = (TP + TN)/(TP + TN + FP + FN) \quad (8)$$

$$\text{Sensitivity} = TP/(TP + FN) \quad (9)$$

$$\text{Specificity} = TN/(FP + TN) \quad (10)$$

where TP (true positive) is the number of instances that actually belong to the class C and are correctly identified by the classifier, FN (false negative) is on the contrary, which is the number of instances that belong to the class C but mistakenly classified. FP (false positive) is the number of instances that do not belong to class C but

Figure 8. The augmented samples of rice disease images.

Table 3. The training results of different approaches on the public rice dataset

Pre-trained model	Ten epochs			Thirty epochs			
	Training accuracy (%)	Validation accuracy (%)	Training loss	Training accuracy (%)	Validation accuracy (%)	Training loss	Validation loss
VGGNet-19	64.76	71.11	0.8347	79.52	74.44	0.6506	0.7124
ResNet-50	62.86	67.78	0.8993	70.48	70.00	0.7364	0.8066
DenseNet-201	84.76	81.11	0.4010	91.90	82.22	0.2436	0.6119
Inception V3	73.33	66.67	0.6169	80.95	71.11	0.4803	1.0351
VGG19-SVM	66.67	80.00	2.0535	72.91	86.67	1.7945	1.4663
Proposed approach	86.19	84.44	0.1430	**92.86**	**92.22**	0.1390	0.1419

Note: Bold values in the table indicates the main significance

incorrectly classified as this classification. TN (true negative) is the number of instances that are not in class *C* in reality, and they are correctly classified. Thus, the models of various CNNs are trained and multiple experiments are conducted on this dataset. Table 3 displays the results of model training for different approaches.

As can be seen in Table 3, the proposed approach outperforms the other well-known CNN architectures on the experimental dataset, even if the optimal classifiers are adopted. The main reason is that the proposed approach makes use of the generic characteristics of images extracted from the ImageNet dataset by DenseNet, and automatically learns specific features of the experimental dataset due to the newly added module. By contrast, the other methods are the individual networks, although the weights are initialized with pre-trained networks instead of random initial weights and the fine-tuning approach are applied, the optimal results of these models are not achieved. Thus, because the proposed approach transfers to the specific task using the pre-trained DenseNet on ImageNet with the Inception module, it combines the advantages of both and achieves the top performances in the experiments of the public dataset. After 30 epochs of training, the training accuracy of the proposed approach achieves 92.86%, the validation accuracy reaches 92.22%, and the validation loss is 0.14, as displayed in Table 3. Furthermore, using the model trained by the proposed approach, the images outside modeling are selected for the class prediction of rice disease images. Figure 9 depicted the receiver operating characteristic (ROC) curve and confusion matrix of detecting results, and the corresponding evaluation indicators are calculated in Table 4.

It can be seen from Table 4 that the average predicting accuracy of the experiment achieves no less than 94.07% on the public dataset, which presents that the proposed DENS-INCEP approach has a significant capability to recognize the rice diseases in the simple background condition. Especially, the specific categories of plant diseases are detected by the proposed approach with the average sensitivity of 91.11%, indicating the validity of the proposed approach.

Experiments on the collected dataset

In the same way, the proposed approach is tested on our own rice leaf disease dataset, which is collected in real-life scenarios with complex background conditions and uneven illumination intensity, such as the surroundings of the field, soils of different colors, and capturing in overcast or sunny weather, etc. The collected rice dataset has 515 images affected by 13 diseases, and each disease class includes originally colored images of varied sizes, about half of which are used for the prediction and evaluation of the model. Particularly, to ensure the diversity of sample images and avoid the overfitting problem, the data augmentation techniques including the color jittering, random rotation, flipping, translation, cropping, and scale transform are applied in the retained images of the dataset, and at least 200 images are guaranteed for each category. Thus, similar to the experiments conducted earlier, the augmented dataset is divided into a training set and a test set with a ratio of 7:3 to determine if the model is overfitting, and related pre-processing works are also performed. The detailed processes are described.
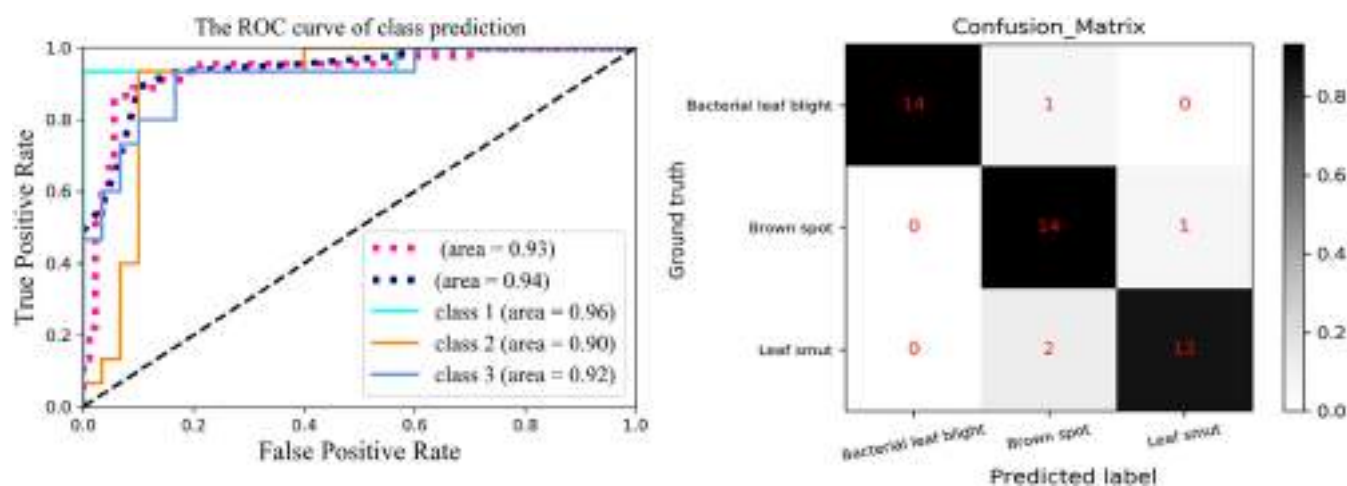
Figure 9. (Left) ROC curve and (right) confusion matrix of test results.

Table 4.	Evaluation indicators of detecting results (%)					
Number	Rice disease classes	Predicted samples	Corrected samples	Accuracy (%)	Sensitivity (%)	Specificity (%)
0	Bacterial leaf blight	15	14	97.78	93.33	100.00
1	Brown spot	15	14	91.11	93.33	90.00
2	Leaf smut	15	13	93.33	86.67	96.67
—	Average	—	—	**94.07**	**91.11**	95.56

(1) Change the size of the images. All the images are resized to the fixed-dimension of 224×224 pixels to fit the model, and with the data augmentation methods, the disease images are augmented to at least 200 images for the categories of small sample data.

(2) Image pre-processing. After resizing the images, the image pre-processing is performed to blacken the shorter sides of the images so that they become the same proportion, thus preserving the information of original images and avoiding image distortion.

(3) Dataset partition. Except for a certain number of validation images, the dataset D is divided into a training set A and testing set B, and thus $D = A + B$.

(4) Model training. Referring to the method proposed earlier, the training set A is applied to train the model. To fully verify the effectiveness, the multiple experiments are conducted with the shuffling of images.

(5) Testing and validation. The testing set B is used to evaluate the model, and the new images outside modeling are applied to verify the effectiveness of the model. The output results are compared with the actual categories and the related evaluation indicators are calculated.

Thus, based on the earlier processes, we perform the model training on our rice leaf disease dataset, and the test accuracies of different approaches are obtained in Table 5. A similar

| Table 5.	The training accuracy and loss of different approaches on our dataset						
	Ten epochs			Thirty epochs			
Pre-trained model	Training accuracy (%)	Validation accuracy (%)	Training loss	Training accuracy (%)	Validation accuracy (%)	Training loss	Validation loss
VGGNet-19	74.86	78.67	0.7874	88.57	82.00	0.4292	0.5447
ResNet-50	49.14	46.00	1.2615	56.86	52.67	1.0628	1.2225
DenseNet-201	84.00	68.00	0.3934	91.14	82.67	0.2672	0.5743
Inception V3	71.71	49.33	0.7794	77.71	60.67	0.6342	1.4707
VGG19-SVM	83.62	79.58	2.1844	90.56	84.38	1.1374	1.3301
Proposed approach	90.00	78.67	0.1881	96.57	90.14	0.1483	0.3059
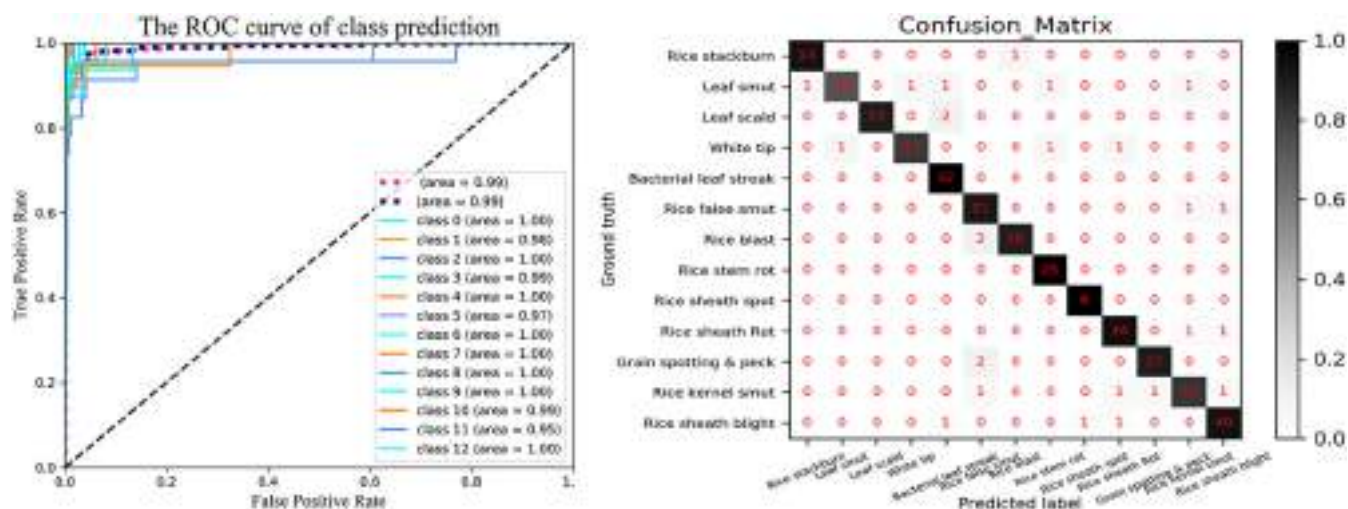
Figure 10. (Left) ROC curve and (right) confusion matrix of rice disease detection.

Table 6. The evaluation indicators of prediction results for different rice disease classes
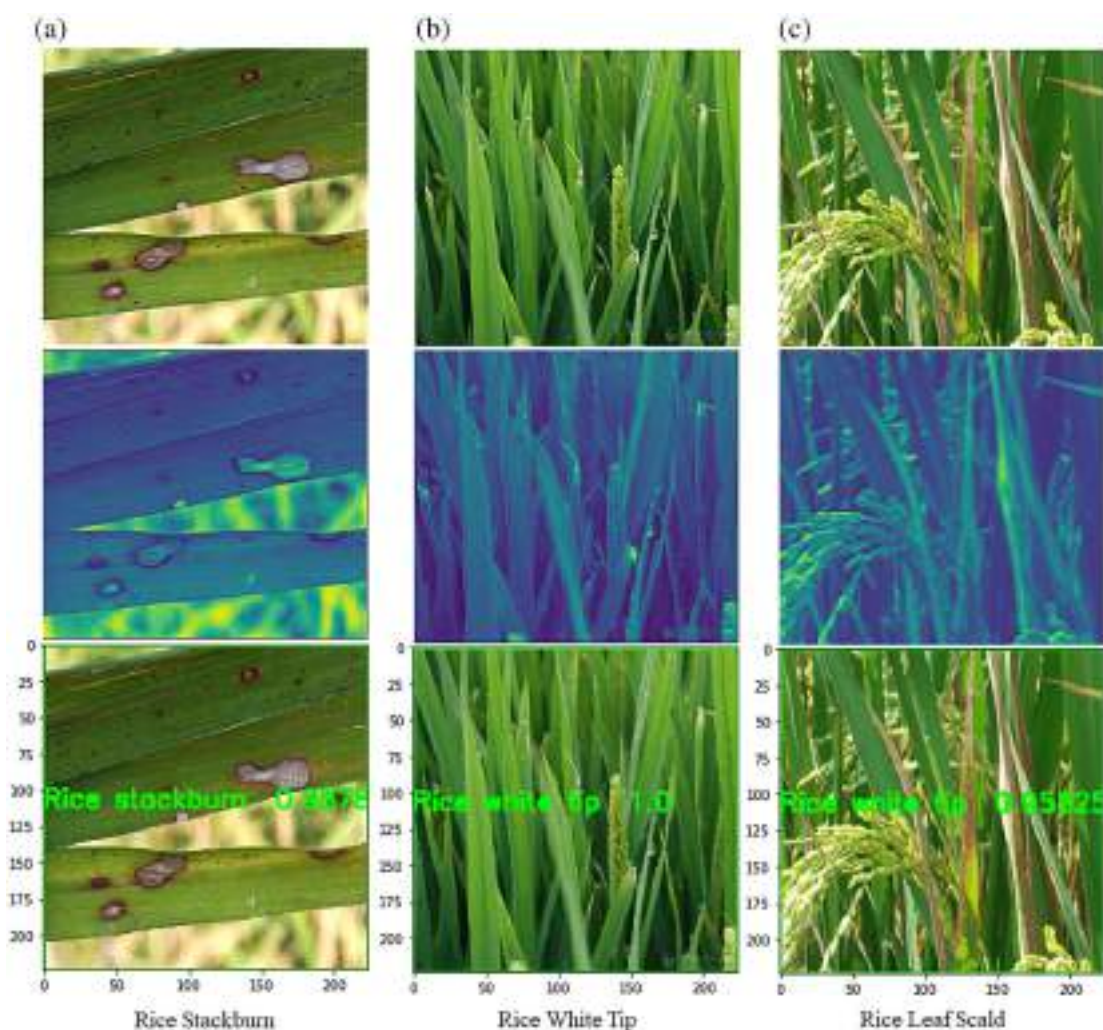
Number	Rice disease classes	Predicted samples	Corrected samples	Accuracy (%)	Sensitivity (%)	Specificity (%)
0	Rice stackburn	14	13	99.32	92.86	99.64
1	Leaf smut	19	14	97.95	73.68	99.63
2	Leaf scald	19	17	99.32	89.47	100.00
3	White tip	16	13	98.63	81.25	**99.64**
4	Bacterial leaf streak	32	32	98.63	100.00	98.46
5	Rice false smut	23	21	97.61	91.30	98.15
6	Rice blast	18	16	98.98	88.89	99.64
7	Rice stem rot	25	25	99.32	100.00	99.25
8	Rice sheath spot	6	6	99.66	100.00	99.65
9	Rice sheath rot	36	34	98.29	94.44	98.83
10	Grain spotting and peck	19	17	98.97	89.47	99.63
11	Rice kernel smut	23	19	97.61	82.61	98.89
12	Rice sheath blight	43	40	97.95	93.02	98.80
—	Average	—	—	98.63	91.12	99.26

conclusion can be drawn from Table 5 that the proposed approach presents a promising result and outperforms the other CNN architectures, which are the state-of-the-art machine learning methods for image recognition. Therefore, after obtaining the trained models, the new unseen images are selected for the class prediction of our method. Figure 10 depicts the ROC curve and confusion matrix of predicting results on our own dataset, and the corresponding evaluation indicators are calculated in Table 6.

As can be observed in Fig. 10(left), the class predictions of the proposed approach show good operating points in the upper, left corner of the ROC curve. For most of the classes, the TPR (true positive rate) is higher when the FPR (false positive rate) is lower, which can be reflected in the confusion matrix of Fig. 10(right) as well. For example, 13 samples are correctly predicted in the class of 'Rice Stackburn' except for one misclassification, and the correct number is 14 for the class prediction of 'Leaf Smut' in 19 samples; there are 17 correctly predicted samples in 19 'Leaf Scald' samples, etc. The average accuracy is achieved 98.63% for the detection of 13 class rice diseases, and the average sensitivity

is realized no less than 91.12% as well, as shown in Table 6. However, individual categories with more misclassifications such as five samples in the class of 'Leaf Smut', which is due to the scenario that some diseases including 'Leaf Scald' and 'White tip' occur with the 'Leaf Smut' in the same training samples and more tiny features are extracted by the classifier, causing the misclassifications. Furthermore, the heterogeneous background condition and uneven illumination intensity also affect the results. Figure 11 displays the partial detection sample images.

As seen in Fig. 11, the top images are the original sample images, the middle images are the extracted feature maps, and the bottom images are the results of class prediction. From Fig. 11, it can be seen that the predicted categories of most samples are consistent with the actual categories of rice images, and most rice diseases are correctly detected by the proposed approach except for the individual samples. For example, the actual category of Fig. 11(a) is the 'Rice Stackburn', which is correctly detected by the proposed approach, and the predicted probability of this rice disease image is greater than 0.98. Likewise, the other sample images such as Fig. 11(b) are also correctly detected by the

Figure 11. The predicted samples of different rice diseases. (a) Rice stackburn, (b) rice white tip, and (c) rice leaf scald.

proposed approach. Although the individual images are misclassified due to some different diseases occurring in the same plant and heterogeneous background conditions, as shown in Fig. 11 (c), most of the rice plant diseases are correctly detected by the proposed approach. Thus, based on the empirical analysis, it can be concluded that the proposed approach has a significant capability for the detection of rice plant diseases and can also be extended to other fields such as online defect detection, target recognition, etc.

CONCLUSION

The timely and effective detection of rice plant diseases is crucial for ensuring the productivity of grain products, hence seeking fast, automatic, less expensive and accurate methods to detect rice disease cases is of great realistic significance. Deep learning techniques, and CNNs, have shown impressive performance in addressing most of the technical challenges associated to plant disease classification. Therefore, in this article, a novel deep learning architecture called DENS-INCEP for rice disease identification is proposed. The pre-trained DenseNet on ImageNet is combined with the Inception module for transfer learning, and the top layers are truncated by defining a new fully-connected Softmax layer with the practical number of classifications. Furthermore, instead

of the original Cross-Entropy Loss function, the Focal Loss function is employed in the network to enhance the learning ability of the tiny lesion features. In this way, the integrated DENS-INCEP enhances the capability of feature extraction and decreases the computational complexity of the model without losing the discriminant formation. Experimental results demonstrated the model with promising performance on both the public dataset and the rice disease image dataset. In future development, we intend to deploy it on mobile devices to monitor and identify the wide range of plant disease information automatically. Meanwhile, we plan to apply it on more real-world applications.

CONFLICT OF INTEREST

The authors declare no conflicts of interest.

REFERENCES

1 Lu Y, Yi S, Zeng N, Liu Y and Zhang Y, Identification of rice diseases using deep convolutional neural networks. *Neurocomputing* **267**: 378–384 (2017).

2 Wang X, Zhang X and Zhou G, Automatic detection of rice disease using near infrared spectra technologies. *J Indian Soc Remote Sens* **45**:785–794 (2017).

3 Al-Hiary H, Bani-Ahmad S, Reyalat M, Braik M and Alrahamneh Z, Fast and accurate detection and classification of plant diseases. *Int J Comput Appl* **17**:31–38 (2011).

4 Bai X, Cao Z, Zhao L, Zhang J, Lv C, Li C *et al.*, Rice heading stage automatic observation by multi-classifier cascade based rice spike detection method. *Agric For Meteorol* **259**:260–270 (2018).

5 Al Bashish D, Braik M and Bani-Ahmad S, Detection and classification of leaf diseases using K-means-based segmentation and. *Inf Technol J* **10**:267–275 (2011).

6 Ghaiwat SN and Arora P, Detection and classification of plant leaf diseases using image processing techniques: a review. *Int J Recent Adv Eng Technol* **2**:1–7 (2014).

7 Yang N, Yuan M, Wang P, Zhang R, Sun J and Mao H, Tea diseases detection based on fast infrared thermal image processing technology. *J Sci Food Agric* **99**:3459–3466 (2019).

8 Wells WM III, Medical image analysis – past, present, and future. *Med Image Anal* **33**:4–6 (2016).

9 Miki Y, Muramatsu C, Hayashi T, Zhou X, Hara T, Katsumata A *et al.*, Classification of teeth in cone-beam CT using deep convolutional neural network. *Comput Biol Med* **80**:24–29 (2017).

10 Gökmen V and Sügüt I, A non-contact computer vision based analysis of color in foods. *Int J Food Eng* **3**:1–13 (2007).

11 Arvis V, Michalet-Doreau B and Debain C, A new methodology to appraise the physical characteristics of maize silage by image processing: effect of the ensiling process. *J Sci Food Agric* **84**:1848–1854 (2004).

12 Mondal S and Bours P, A study on continuous authentication using a combination of keystroke and mouse biometrics. *Neurocomputing* **230**:1–22 (2017).

13 Lee SJ, Yun JP, Koo G and Kim SW, End-to-end recognition of slab identification numbers using a deep convolutional neural network. *Knowl-Based Syst* **132**:1–10 (2017).

14 Kien BH, Iba D, Ishii Y, Tsutsui Y, Miura N, Iizuka T *et al.*, Crack detection of plastic gears using a convolutional neural network pre-learned from images of meshing vibration data with transfer learning. *Forsch Ingenieurwes* **83**:645–653 (2019).

15 Hung JC, Lin KC and Lai NX, Recognizing learning emotion based on convolutional neural networks and transfer learning. *Appl Soft Comput* **84**:105724 (2019).

16 Anitha U, Malarkkan S, Premalatha J and Prince PGK, Study of object detection in sonar image using image segmentation and edge detection methods. *Indian J Sci Technol* **9**:42 (2016).

17 Duan Y, Liu F, Jiao L, Zhao P and Zhang L, SAR image segmentation based on convolutional-wavelet neural network and markov random field. *Pattern Recogn* **64**:255–267 (2017).

18 Zhang L, Mistry K, Neoh SC and Lim CP, Intelligent facial emotion recognition using moth-firefly optimization. *Knowl-Based Syst* **111**: 248–267 (2016).

19 Ebrahimi MA, Khoshtaghaza MH, Minaei S and Jamshidi B, Vision-based pest detection based on SVM classification method. *Comput Electron Agric* **137**:52–58 (2017).

20 García J, Pope C and Altimiras F, A distributed-means segmentation algorithm applied to lobesia botrana recognition. *Complexity* **2017**:14 (2017).

21 Singh AK, Rubiya A and Raja BS, Classification of rice disease using digital image processing and svm classifier. *Int J Electr Electron Eng* **7**: 294–299 (2015).

22 Phadikar S, Sil J and Das AK, Classification of rice leaf diseases based on morphological changes. *Int J Inf Electron Eng* **2**:460–463 (2012).

23 Kahar MA, Mutalib S and Abdul-Rahman S, Early detection and classification of paddy diseases with neural networks and fuzzy logic. In *Proceedings of the 17th International Conference on Mathematical and Computational Methods in Science and Engineering, MACMESE* **2015**: 248–257 (2015).

24 Orillo JW, Cruz JD, Agapito L, Satimbre PJ and Valenzuela I, Identification of diseases in rice plant (*Oryza sativa*) using back propagation artificial neural network. In *2014 International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)* **2014**: 1–6, IEEE.

25 Barbedo JG, Factors influencing the use of deep learning for plant disease recognition. *Biosyst Eng* **172**:84–91 (2018).

26 Kamilaris A and Prenafeta-Boldú FX, Deep learning in agriculture: a survey. *Comput Electron Agric* **147**:70–90 (2018).

27 Sardogan M, Tuncer A and Ozen Y, Plant leaf disease detection and classification based on CNN with LVQ algorithm. In *2018 3rd International Conference on Computer Science and Engineering (UBMK)* **2018**: 382–385, IEEE.

28 Barbedo JGA, Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification. *Comput Electron Agric* **153**:46–53 (2018).

29 Ferentinos KP, Deep learning models for plant disease detection and diagnosis. *Comput Electron Agric* **145**:311–318 (2018).

30 Mohanty SP, Hughes DP and Salathé M, Using deep learning for image-based plant disease detection. *Front Plant Sci* **7**:1419 (2016).

31 Shrivastava VK, Pradhan MK, Minz S and Thakur MP, Rice plant disease classification using transfer learning of deep convolution neural network. *Int Arch Photogramm Remote Sens Spatial Inf Sci* **XLII-3/W6**: 631–635 (2019).

32 Ma J, Du K, Zheng F, Zhang L, Gong Z and Sun Z, A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. *Comput Electron Agric* **154**:18–24 (2018).

33 Qiang Z, He L and Dai F, Identification of plant leaf diseases based on inception V3 transfer learning and fine-tuning, in *International Conference on Smart City and Informatization*, Vol. **1122**. Springer, Singapore, pp. 118–127 (2019).

34 Kawasaki Y, Uga H, Kagiwada S and Iyatomi H, Basic study of automated diagnosis of viral plant diseases using convolutional neural networks, in *International Symposium on Visual Computing*, Vol. **2015**. Springer, Cham, pp. 638–645 (2015).

35 Szegedy C, Vanhoucke V, Ioffe S, Shlens J and Wojna Z, Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition* 2016: 2818–2826 (**2016**).

36 Huang G, Liu Z, Van Der Maaten L and Weinberger KQ, Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* **2017**:4700–4708 (2017).

37 Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S *et al.*, Imagenet large scale visual recognition challenge. *Int J Comput Vision* **115**: 211–252 (2015).

38 Lumini A and Nanni L, Deep learning and transfer learning features for plankton classification. *Ecol Inform* **51**:33–43 (2019).

39 Kessentini Y, Besbes MD, Ammar S and Chabbouh A, A two-stage deep neural network for multi-norm license plate detection and recognition. *Expert Syst Appl* **136**:159–170 (2019).

40 Ghazi MM, Yanikoglu B and Aptoula E, Plant identification using deep neural networks via optimization of transfer learning parameters. *Neurocomputing* **235**:228–235 (2017).

41 Lin TY, Goyal P, Girshick R, He K and Dollár P, Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision* **2017**: 2980–2988 (2017).

42 Keras-GPU. Available online: https://anaconda.org/anaconda/keras-gpu [accessed 17 June, 2019].

43 GeForce GTX 1060. Available online: https://www.nvidia.com/en-us/geforce/products/10series/geforce-gtx-1060/specifications [accessed 17 June, 2019].

44 Blake CL and Merz CJ, UCI repository of machine learning database. https://archive.ics.uci.edu/ml/index.php (1998).

45 Prajapati HB, Shah JP and Dabhi VK, Detection and classification of rice plant diseases. *Intell Decis Technol* **11**:357–373 (2017).

46 Dosovitskiy A, Fischer P, Springenberg JT, Riedmiller M and Brox T, Discriminative unsupervised feature learning with exemplar convolutional neural networks. *IEEE Trans Pattern Anal Mach Intell* **38**: 1734–1747 (2015).

47 Simonyan K and Zisserman A, Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv*: 1409.1556 (2014).

48 He K, Zhang X, Ren S and Sun J, Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* **2016**: 770–778 (2016).