IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 12, 2018, accepted December 27, 2018, date of publication January 10, 2019, date of current version February 4, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2891739

A Unified Matrix-Based Convolutional Neural Network for Fine-Grained Image Classification of Wheat Leaf Diseases

ZHONGQI LIN[1,2], (Graduate Student Member, IEEE), SHAOMIN MU[3], FENG HUANG[2], KHATTAK ABDUL MATEEN[1,4], MINJUAN WANG[1,2], (Member, IEEE), WANLIN GAO[1,2], AND JINGDUN JIA[2]

[1]College of Information and Electrical Engineering, China Agricultural University, Beijing 100083, China
[2]Key Laboratory of Agricultural Informatization Standardization, Ministry of Agriculture and Rural Affairs, Beijing 100083, China
[3]College of Information Science and Engineering, Shandong Agricultural University, Taian 271018, China
[4]Department of Horticulture, Agricultural University Peshawar, Peshawar 25120, Pakistan

Corresponding authors: Wanlin Gao (wanlin_cau@163.com) and Jingdun Jia (jiajd@most.cn)

This work was supported in part by the Project of Scientific Operating Expenses, Ministry of Education of China, under Grant 2017PT19, in part by the China Postdoctoral Science Foundation under Grant 2018M630222, in part by the National Natural Science Foundation for the Youth of China, Natural Science Foundation of Shandong Province, under Grant ZR2018QF002, in part by the Provincial Project, Department of Science and Technology of Shandong Province, under Grant GGGX109002, and in part by the Natural Science Foundation of Shandong Province, Natural Science Foundation of Shandong Province, under Grant ZR2012FM024.

ABSTRACT Fine-grained image classification methods often suffer from the challenge that the subordinate categories within an entry-level category can only be distinguished by subtle differences. Crop disease classification is affected by various visual interferences, including uneven illumination, dew, and equipment jitter. It demands an effective algorithm to accurately discriminate one category from the others. Thus, the representational ability of algorithm needs to be strengthened to learn a robust domain-specific discrimination through an effective way. To address this challenge, a unified convolutional neural network (CNN) denoting the matrix-based convolutional neural network (M-bCNN) was proposed. Its hallmark is the convolutional kernel matrix, whose convolutional layers are arranged parallelly in the form of a matrix, and integrated with DropConnect, exponential linear unit, local response normalization, and so on to defeat over-fitting and vanishing gradient. With a tolerable addition of parameters, it can effectively increase the data streams, neurons, and link channels of the model compared with the commonly used plain networks. Therefore, it will create more non-linear mappings and will enhance the representational ability with a tolerable growth of parameters. The images of winter wheat leaf diseases were utilized as experimental samples for their strong similarities among sub-categories. A total of 16 652 images containing eight categories were collected from Shandong Province, China, and were augmented into 83 260 images. The M-bCNN delivered significant improvements and achieved an average validation accuracy of 96.5% and a testing accuracy of 90.1%; this outperformed AlexNet and VGG-16. The M-bCNN demonstrated accuracy gains with a convolutional kernel matrix in fine-grained image classification.

INDEX TERMS Convolutional neural network, fine-grained image classification, deep learning, convolutional kernel matrix, wheat leaf diseases.

I. INTRODUCTION

Early and accurate detection and diagnosis of plant diseases are key factors in wheat production and for reduction of both qualitative and quantitative losses in crop yield [1]. Therefore, developing technologies to accurately classify the categories of wheat leaf diseases is crucial for disease prevention. The state-of-the-art advancements in artificial intelligence and computer vision domains have actually motivated researchers to employ this effective technology in agriculture for automatic categorization of crop diseases caused by biotic and abiotic stresses [2]–[5]. Although remarkable performances have been achieved in normal diseases classification, it is still hard to distinguish several diseases with subtle discrimination.

Fine-grained image classification aims at discriminating the sub-categories sharing one common basic-level category through digital images [6], e.g., classifying different vehicle makes and models [7]–[10], tree categories [11], bird classes [12]–[14], dogs classes [15]–[17], flower species [18], aircrafts [19], [20], body parts [21], [22], etc. Due to its tremendous challenge and study merit both in theory and practice, fine-grained image classification has been extensively studied recently: Liang *et al.* [23] proposed a Gaussian mixture model, which fused local features by Gaussian mixture layer and achieved high classification accuracy; Iscen *et al.* [24] adopted approaches based on superpixels, edges, and a bank of Zernike filters used as detectors, and they found that a better accuracy was achieved when the patches were extracted along the edges and not around the detected regions; Xuan *et al.* [25] proposed a novel evolving convolutional neural network (ECNN), which could use the limited clearly labeled images and weakly labeled images for better fine-grained classification of CIFAR-10, Oxford pets, etc; Seo and Shin [26] proposed to pre-train the GoogLeNet on ImageNet dataset and fine-tune fine-grained fashion dataset based on design attributes apparel classification, and their strategy got promising performance; Zhang *et al.* [27] proposed a fine-grained vehicle recognition framework based on lightweight convolutional neural network (CNN) with combined learning strategy, and competitive recognition performances were achieved whilst decreasing the computational complexity; Zhang *et al.* [28] developed a novel fine-grained image categorization system based on an active learning algorithm and support vector machine (SVM), which achieved better spatial pyramid matching performance and categorization accuracy.

Through the above investigation, we learned that most previous works were aimed at boosting up the classification rate from three main aspects:

1) more precise location of object and domain, which is also known as the global/domain level attention.
2) more robust feature representations for subordinate categories discrimination.
3) human in the loop [29] and reinforcement learning.

Since our goal is automatic fine-grained image classification and our design is based on a simple intuition, i.e. directly boosting up the accuracy through more robust discriminative features extracted by an effective algorithm, we are more focused on the related research of the first two, Due to small discrepancies, different sub-categories are always distinguished by domain-specific areas, such as the texture of a feather [30]–[34] and a petal [35], [36], the color of a coat [37], [38] and a beak [39]–[43], the shape of a trademark [35], [45] and a vehicle [46], [47], etc. Consequently, detecting these subtle discriminative domains from similar areas is crucial for fine-grained image classification [48]–[52].

Another point is that the fine-grained classification tasks are common and more challenging in uncontrolled realistic crop disease classification. Different subordinate categories have almost similar appearance of diseases. Their discriminations mainly exist in subtler areas. Perhaps counter intuitively, intra-category discriminations can be very larger than inter-category among different sub-categories in some cases, as depicted in **Fig.** 1. More seriously, a number of visual interferences, such as reflection, dispersion, and blur caused by dew, equipment jitter, and extreme illuminations can lead to large visual interferences in terms of leaf appearance. This can severely degrade the accuracy and robustness of the disease classification [53], [54]. Our previous work [55] also suffered these interferences seriously.

Due to the adequate collected samples and research value both in theory and practice, our work focuses on the fine-grained image classification of winter wheat leaf diseases. Traditionally, early classification of wheat leaf disease is dominated by delicate hand-crafted features [56]–[61],

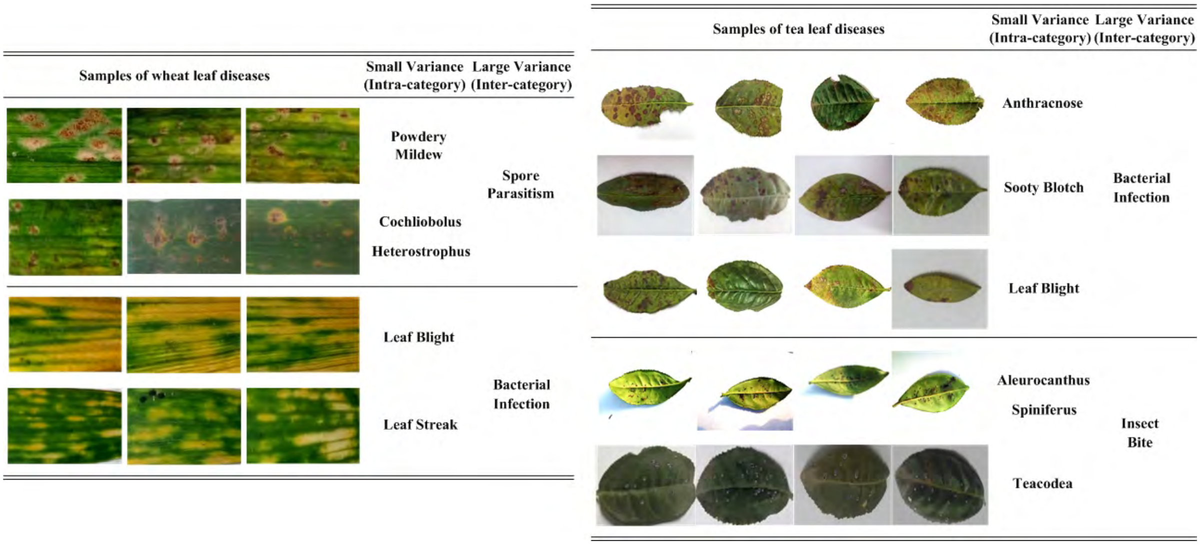


FIGURE 1. Illustration of the difficulty of fine-grained image classification of crop diseases: Large intra-category discriminations and small inter-category discriminations. (a) The sub-categories and basic-level categories of wheat leaf diseases images. (b) The sub-categories and basic-level categories of tea leaf diseases images.

e.g., HOG, SIFT, SURF, and LBP. However, the design of these descriptors is typically time-consuming and their performances are unsatisfactory [23]. Thus, researchers have attempted to tackle the problems by proposing artificial intelligence methods. For instance, Zhao *et al.* [62] proposed an optimized MSF-AdaBoost model to classify and monitor powdery mildew on winter wheat on a regional scale. A high classification accuracy and promising monitoring performance was achieved; Tian *et al.* [63] presented an SVM-based Multiple Classifier System (MCS) for pattern recognition of wheat leaf diseases. Compared with the previous classifiers, their algorithm could achieve better recognition rate; Niu *et al.* [64] proposed a modified K-means clustering for efficient identification of wheat leaf diseases, and better performance was achieved for three common diseases (powdery mildew, leaf rust, and stripe rust); Yang *et al.* [65] presented a diagnosis model of *stripe rust* in field scales based on Bayesian network, which provided technical support for accurate identification and short-term prediction of stripe rust on a small scale.

The above methods are concerning to surface learning. Although some progress has been made, there is still some room and potential for improvement. The extraction of handcrafted features, such as inertia moment, roundness, and entropy largely relies on prior knowledge; thus, the extracted features are often inadequate and lacking in detail [55]. Furthermore, while shallow-level features can be extracted effortlessly, abstract representations hidden in the deeper level are difficult to obtain without learning procedures [55].

Moreover, the main challenge for fine-grained image classification of wheat leaf disease is indubitably the very small discrepancies among different categories. Specifically, the difficulties mainly come from three aspects:

1) the strong similarity among different disease spots.
2) the large visual interferences of the cluttered environments.
3) the large search space of possible disease spot positions.

Consequently, an effective classification model was required to accurately extract subtle features from the domain-regions. The model that would have a high objectiveness, containing the vital discriminates of certain objects. Driven by this requirement, an improved convolutional neural network codenamed matrix-based convolutional neural network (M-bCNN) has been proposed in this work.

CNN is a multilayer variant perceptron (MLP) [66] inspired by Hubel-Wiesel biological vision system. It can adaptively construct implicit feature description through multi-layer non-linear mapping under training data driving [67]. During the last seven years, mainly due to the state-of-the-art performance of CNN, the quality of image classification and other related fields have progressed at a dramatic pace. In 2012, Krizhevsky *et al.* [68] won two first prizes (in two separate tasks) for developing AlexNet model in ImageNet Large Scale Visual Recognition Competition (ILSVRC) [69], where its accuracy rate exceeded by 10% that of the second-placed competitor. In 2014, two milestones in face classification were achieved, when Taigman *et al.* [70] and Ouyang *et al.* [71] proposed CNN based DeepFace and DeepID. These proved to be the most perfect authentication models for face classification. Two other brands of CNN frameworks with the design concept of "go deeper" became the champion (GoogLeNet [72]) and runner-up (Visual Geometry Group (VGG) [73]) in ILSVRC-2014. The following year, ResNet [74] designed by Microsoft Research Asia (MSRA) won the championship in ILSVRC-2015. Its Top-5 error rate (3.57%) was lower than humans' classification error rate (5.1%), which illustrated that its object recognition ability surpassed that of human eye. In 2016, DeepMind, a subsidiary of Google, applied CNN to an intelligent robot AlphaGo [75] and defeated Go champion Li Shishi. Sharing the congenital advantages of CNN (though it is hard to see the study of CNN in the fine-grained image classification of wheat leaf diseases), we proposed to utilize it as the theoretical basis in this work.

However, we needed to enhance the representational ability of CNN to better tackle the fine-grained image classification tasks. It is obvious that most representative CNN models gain improvement in accuracy through stacked layers [76], [77]. During 2012 to 2015, all the leading works [68], [72], [74] in the challenging ILSVRC [69] exploited "extreme deep" models, with a depth of 8 [68] to 152 [74]. He *et al.* [74] even utilized ResNet with 1202 layers to analyze CIFAR-10. Recent evidences reveal that increasing hidden layers is essential for success in the current state-of-the-art convolutional networks [73], [74], [78], [79]. Previous studies show that most deep neural networks typically follow a standard structure originating from LeNet-5 – linearly stacked convolutional layers are optionally followed by one or more subpooling layers and fully-connected layers. These "plain" nets that simply stack layers [74] prevail in the image classification literature and have produced impressive results so far on CIFAR-10, MNIST and other classification tasks. Nevertheless, the design is not efficient to improve the representational ability of networks compared with parallel structure, which is concretely embodied in the growth rate of parallelism level, data streams, scheduling efficiency, neurons, link channels, etc [72], [74]. Moreover, if several convolutional layers are linearly chained together, this will result in a quadratic increase of parameters and computational budget. Then more terrible over-fitting and curse of dimensionality will follow, which will result in serious issues with accuracy [72], [74], [79].

The depth of model representations is also imparative for distinguishing fine-grained visual categories. In order to find an effective method to increase representational ability without obvious side-effects, an improved hierarchical CNN denoting the M-bCNN was proposed and its gratifying performances were evaluated in fine-grained image classification of wheat leaf diseases.

In order to employ the discriminative feature representational ability of CNN, our model originates as a fundamental plain network (i.e. AlexNet [68]). First, we add two convolutional neural layers to the first two low layers (i.e. *conv_1* and *conv_2*) for extracting the global features of images. Then the next three linearly stacked higher convolutional neural layers (i.e. *conv_3*, *conv_4*, and *conv_5*) take place by three 3 × 3 convolutional kernel matrixes. These are responsible for enhancing representational ability and searching for domain-specific representations in subtler areas. Meanwhile, other tricks, such as exponential linear unit (ELU), local response normalization (LRN), and DropConnect are also integrated together to inhibit vanishing gradient and over-fitting. The overall design conception obeys the following principle: extracting the global features and then searching for domain-specific discrimination.

Due to the large parameters of the model, direct training from scratch on the image set of wheat leaf diseases will result in over-fitting. Therefore, M-bCNN is first pre-trained on the ImageNet database [69]. Based on the training weights, we fine-tuned the model to adapt to the fine-grained images. M-bCNN has proven theoretically and practically that the convolutional kernel matrix is effective to increase the number of data streams, neurons, and link channels, while it inhibits parameter growth. In addition, the tiny features can be extracted by minitype convolution filters arranged in matrix. Moreover, they can combine freely with each other, because of fully-connected mode, generating different feature maps thus improving the flexibility and characterization ability of the model. Therefore, convolutional kernel matrix caters for the strict requirements of enhancing representational ability and suppressing time complexity. This is meaningful for fine-grained image classification of crop diseases. Convolutional kernel matrix allows M-bCNN to acquire the accuracy gains from increased hidden layers without obvious accuracy loss penalty, producing results which significantly compete against AlexNet and VGG-16.

The main novelties and contributions of this research are summarized in three points:

1) To the best of our knowledge, it is the first work that used improved CNN for the fine-grained image classification of wheat leaf diseases. A novel hybrid CNN structure codenamed M-bCNN is proposed, which significantly increases the data streams, neurons, and link channels. The effectiveness of its hallmark, the convolutional kernel matrix, is proven both theoretically and practically.
2) As there is no large-scale publicly available image set of wheat leaf diseases at present, a total of 16,652 high-fidelity winter wheat leaf images, containing eight categories, were collected from locations of Shandong province, China. Moreover, a total of 83,260 augmented images were produced by five augmentation methods. Obviously, this is the first large-scale high-resolution image set of winter wheat leaf diseases. We intend to open source this set when it gets richer both in terms of quantity and species.
3) We have undertaken a significant amount of work on the image set. Our proposed model achieves higher validation accuracy, individual classification rate, precision, recall, and F1-score improvement with a tolerable parameter addition.

The remainder of this paper is as follows. Section II illustrates the methodology of the proposed M-bCNN and the effectiveness of convolutional kernel matrix. Section III presents the construction of the original and augmented image sets of wheat leaf diseases. The details of experimental process and results are being covered in Section IV. The detailed discussion and analysis are elaborated in Section V. Finally, in Section VI, the concluding remarks and suggestions for future works are provided.

II. MATRIX-BASED CONVOLUTIONAL NEURAL NETWORK

A. OVERVIEW OF M-bCNN

In an attempt to leverage the success of CNN for object classification, the M-bCNN for fine-grained image classification of wheat leaf disease is proposed. This section details the novelty of our method. It describes the new hierarchical M-bCNN architecture that integrates the proposed convolutional kernel matrix and other tricks, such as ELU, LRN, and DropConnect. Convolutional kernel matrix aims at increasing the model's representational ability so as to learn a domain-specific discrimination to deal with fine-grained classification, whilst suppressing parameter growth rate. The model depicted in **Fig.2** is called as M-bCNN-CKM-3 for its 3 × 3 convolutional kernel matrix.

As obvious from **Fig.** 2, M-bCNN-CKM-3 mainly contains four convolutional layers (Conv$_1$, Conv$_2$, Conv$_3$, Conv$_4$), three MaxPooling layers (S$_2$, S$_4$, S$_7$), three 3 × 3 convolutional kernel matrixes (CKM-3$_5$, CKM-3$_6$, CKM-3$_7$), and three fully-connected layers (F$_8$, F$_9$, F$_{10}$). Specifically, CKM-3$_5$, CKM-3$_6$, and CKM-3$_7$ are responsible for increasing the model depth and representational ability. Each one contains nine 3×3 convolutional layers (Conv$_{(1,1)}$, Conv$_{(2,1)}$, Conv$_{(3,1)}$; Conv$_{(1,2)}$, Conv$_{(2,2)}$, Conv$_{(3,2)}$; Conv$_{(1,3)}$, Conv$_{(2,3)}$, Conv$_{(3,3)}$) and each layer contains 96 3 × 3 convolutional kernels. See **Fig.** 3 for detailed architecture and data streams in convolutional kernel matrix.

1) DropConnect

Fig. 3 reveals that the input pixel vector $x = [x_1, x_2, x_3, \ldots, x_n]$ is first processed by DropConnect. Since the abundant training parameters and complex structures in convolutional kernel matrix easily cause over-fitting, DropConnect is utilized not only in fully-connected layers, but also in CKM-3$_5$, CKM-3$_6$, and CKM-3$_7$. It can randomly mask the weights of convolution kernels through a binary matrix shown in Eq. (1). The model tends to be less sensitive to the specific weights of neurons, hence less likely to overfit the training samples and capable of better generalization ability.

$$\begin{cases} x_i * M_i \\ M_i \sim Bernoulli\ distribution\ (p) \end{cases} \tag{1}$$

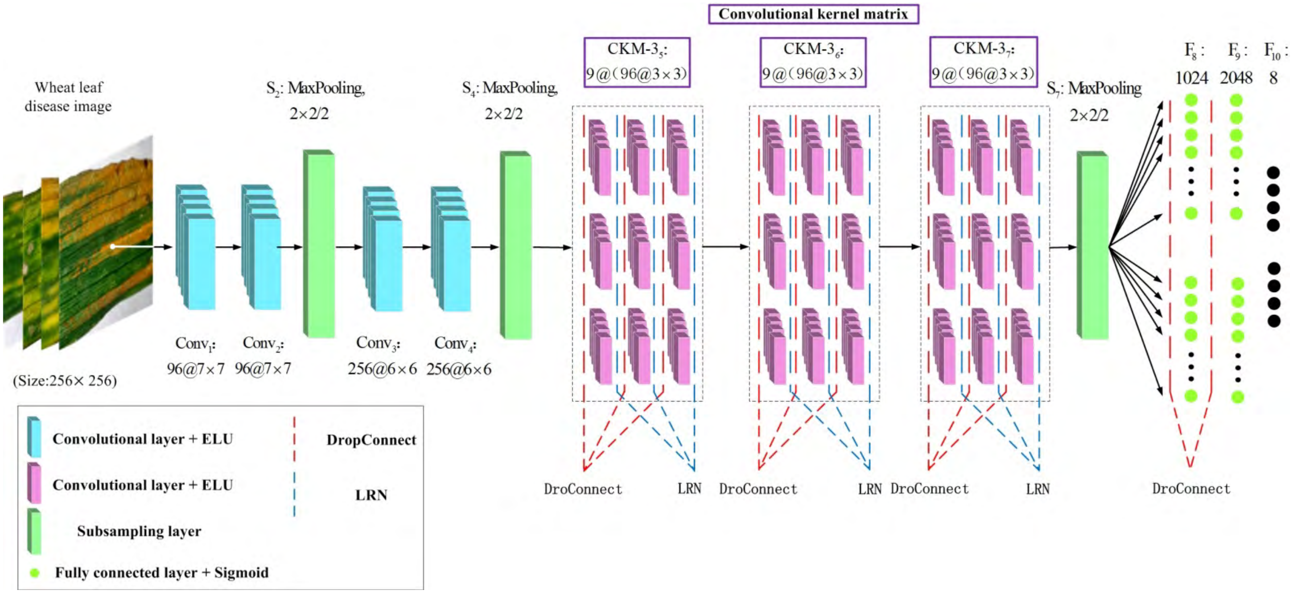


FIGURE 2. The overview of the proposed M-bCNN-CKM-3. Layers of original CNN are shown in green and blue. Convolutional kernel matrixes that we propose are in purple.

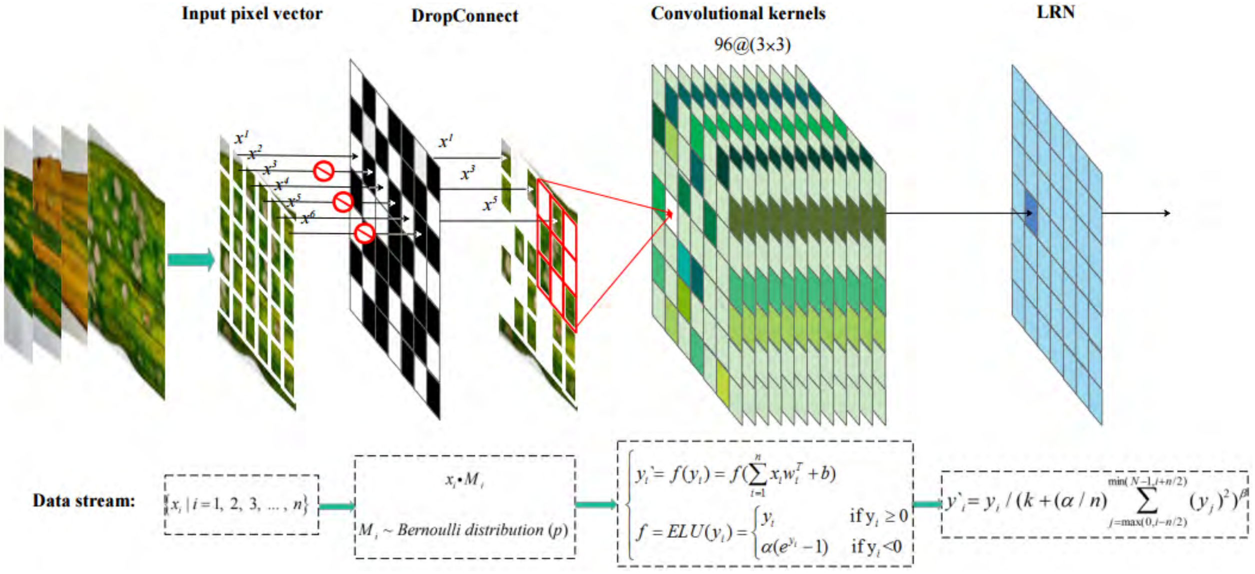

Input pixel vector
DropConnect
Convolutional kernels
96@(3×3)
LRN
x¹
x²
x³
x⁴
x⁵
x⁶
x¹
x³
x⁵
Data stream:
{xᵢ | i = 1, 2, 3, ..., n}
xᵢ • Mᵢ
Mᵢ ~ Bernoulli distribution (p)
yᵢ' = f(yᵢ) = f(Σᵢ₌₁ⁿ xᵢwᵢᵀ + b)
f = ELU(yᵢ) = { yᵢ if yᵢ ≥ 0 ; α(e^yᵢ − 1) if yᵢ < 0
y'ᵢ = yᵢ / (k + (α / n) Σ_{j=max(0,i−n/2)}^{min(N−1,i+n/2)} (yⱼ)²)^β

FIGURE 3. Data streams in convolutional kernel matrix.

where x_i is the input signal, M_i represents a binary random mask matrix, which obeys the Bernoulli distribution. The DropConnect rate increases from 0.15 to 0.5.

2) EXPONENTIAL LINEAR UNIT (ELU)

Next, the processed feature maps are calculated by convolution filters. In order to inhibit vanishing gradient and increase model convergence rate, ELU is utilized as the activation function in convolutional layers, convolutional kernel matrixes and subsampling layers. Suppose the input signals are denoted as $x = [x_1, x_2, x_3, \ldots, x_n, b_i]^T$, the data streams in convolution filter are depicted as below:

$$\begin{cases} y_i` = f(y_i) = f(\sum_{i=1}^{n} x_i w_i^T + b) \\ f = ELU(y_i) = \begin{cases} y_i & \text{if y}_i \geq 0 \\ \alpha(e^{y_i} - 1) & \text{if y}_i < 0 \end{cases} \end{cases} \tag{2}$$

where x_i and y_i` represent the input signals and output feature maps respectively; f represents the non-linear activation function whose role is played by ELU, α is initialized to 0.25 and then self-adjusted by optimization;

$w = [w_1, w_2, w_3, \ldots, w_n, +1]^T$ denotes the weights of ith convolution filter, b is the bias.

3) LOCAL RESPONSE NORMALIZATION (LRN)

After the non-linear mapping of ELU, we employ the channel internal normalization contained in LRN for better generalization ability. Its local region is extended in the independent channel. The received signal is normalized as shown in Eq. (3).

$$y_i'' = y_i'/(k + (\frac{\alpha}{n}) \sum_{j=\max(0,i-n/2)}^{\min(N-1,i+n/2)} (y_j')^2)^\beta \qquad (3)$$

where y_i' and y_i'' represent the input and output feature maps of LRN respectively; α and β denote the scaling factor and exponential term respectively; N and n represent the number of channels and local size of the normalized range respectively. The variables α, β, and n are initialized to 0.0001, 0.75, and 5 respectively, following Krizhevsky *et al.* [68].

Finally, M-bCNN ends with an eight-way fully-connected layer with Softmax [81], [82] function:

$$S_i = \frac{e^{V_i}}{\sum_{j=1}^{K} e^{V_j}} \qquad (4)$$

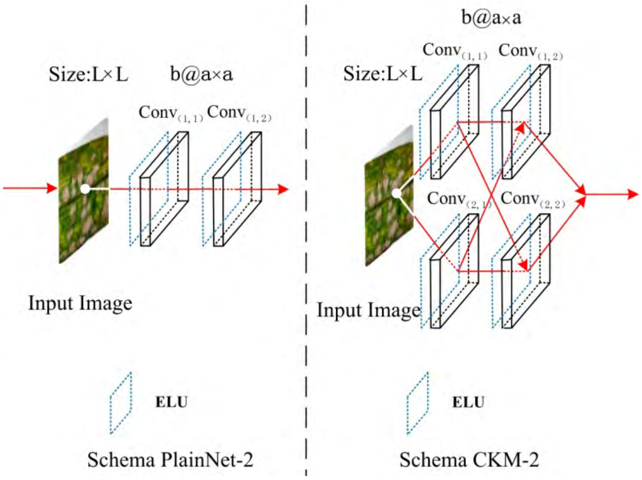
where e^{V_i} and e^{V_j} represent the probability belonging to i and j categories respectively; k denotes the number of categories and it is initialized into eight in this paper. The prediction of each category can be calculated by Softmax function.

B. CONVOLUTIONAL KERNEL MATRIX

In this section, we demonstrate the positive effect of convolutional kernel matrix on representational ability enhancing and parameter growth inhibition. The commonly-used plain nets and proposed convolutional kernel matrix are compared in terms of their structures, data streams, neurons, link channels and training parameters. This is because they largely reflect the performance of a neural network from a mathematical point of view.

1) SCHEMA PlainNet-2 AND SCHEMA CKM-2

In **Fig.** 4, we hypothesize that the size of input image is L × L. Convolutional layers Conv$_{(1,1)}$ and Conv$_{(1,2)}$ both consist of b a × a convolution filters in Schema PlainNet-2. It represents a standard and common CNN structure, called plain net, starting from LeNet-5 - linearly stacked convolutional layers are optionally followed by one or more normalization layers, max-pooling and fully-connected layers. Based on the serial structure of Schema PlainNet-2, we turn the network into its matrix version. In Schema CKM-2, the 2 × 2 convolutional kernel matrix is made up of four convolutional layers (Conv$_{(1,1)}$, Conv$_{(1,2)}$, Conv$_{(2,1)}$, Conv$_{(2,2)}$) and each one is composed of b a×a convolution filters. Specifically, Conv$_{(1,1)}$ and Conv$_{(2,1)}$ are fully connected to Conv$_{(1,2)}$, and Conv$_{(2,2)}$. The data streams of Schema PlainNet-2 and Schema CKM-2 are shown in **Table** 1.



Schema PlainNet-2

Schema CKM-2

FIGURE 4. The structure of Schema PlainNet-2 and Schema CKM-2. Left: a 2-layer plain network as a reference. Right: a 2 × 2 convolutional kernel matrix.

TABLE 1. Data streams in Schema PlainNet-2 and Schema CKM-2.

Name	Data Stream
Schema PlainNet-2	Conv$_{(1,1)}$ →Conv$_{(1,2)}$
Schema CKM-2	Conv$_{(1,1)}$ → Conv$_{(1,2)}$, Conv$_{(2,1)}$ → Conv$_{(1,2)}$, Conv$_{(1,1)}$ → Conv$_{(2,2)}$, Conv$_{(2,1)}$ → Conv$_{(2,2)}$.

The number of neurons, link channels, and training parameters of the two schemas are shown in **Table** 2.

Table 1 reveals that the number of data streams in Schema CKM-2 is four times that of Schema PlainNet-2, which provides more pipelines for feature integration. Accordingly, the number of link channels in Schema CKM-2 is four times that of Schema PlainNet-2 in **Table** 2, which brings more non-linear mappings for feature extraction. In addition, the number of neurons in Schema CKM-2 is two times that of Schema PlainNet-2. It means stronger feature extraction ability. The increase of neurons and link channels are both meaningful for boosting the model's representational ability. Meanwhile, the number of training parameters is also increased with the addition of layers, but not enough to cause serious computational burden.

2) SCHEMA PlainNet-3 AND SCHEMA CKM-3

In **Fig.** 5, linearly stacked convolutional layers, Conv$_{(1,1)}$, Conv$_{(1,2)}$ and Conv$_{(1,3)}$ consist of b a × a convolution filters in Schema PlainNet-3. In Schema CKM-3, convolutional kernel matrix (3 × 3) is made up of nine convolutional layers (Conv$_{(1,1)}$, Conv$_{(2,1)}$, Conv$_{(3,1)}$; Conv$_{(1,2)}$, Conv$_{(2,2)}$, Conv$_{(3,2)}$; Conv$_{(1,3)}$, Conv$_{(2,3)}$, Conv$_{(3,3)}$) and each one is also made up of b a × a convolution filters. Layers in adjacent columns are fully connected with each other. Therefore, there is nine data streams in Schema CKM-3. See **Table** 3 for detailed data streams in Schema PlainNet-3 and Schema CKM-3.

TABLE 2. The number of neurons, link channels and training parameters in Schema PlainNet-2 and Schema CKM-2.

Name	Number of Neurons	Number of Link Channels	Number of Training Parameters
Schema PlainNet-2	$b[(L-a+1)^2+(L\text{-}2a+2)^2]$	$(a^2+1)b[(L\text{-}a\text{+}1)^2+(L\text{-}2a\text{+}2)^2]$	$2(a^2+1)b$
Schema CKM-2	$2b[(L-a+1)^2+(L\text{-}2a+2)^2]$	$4(a^2+1)b[(L-a+1)^2+(L-2a+2)^2]$	$4(a^2+1)b$

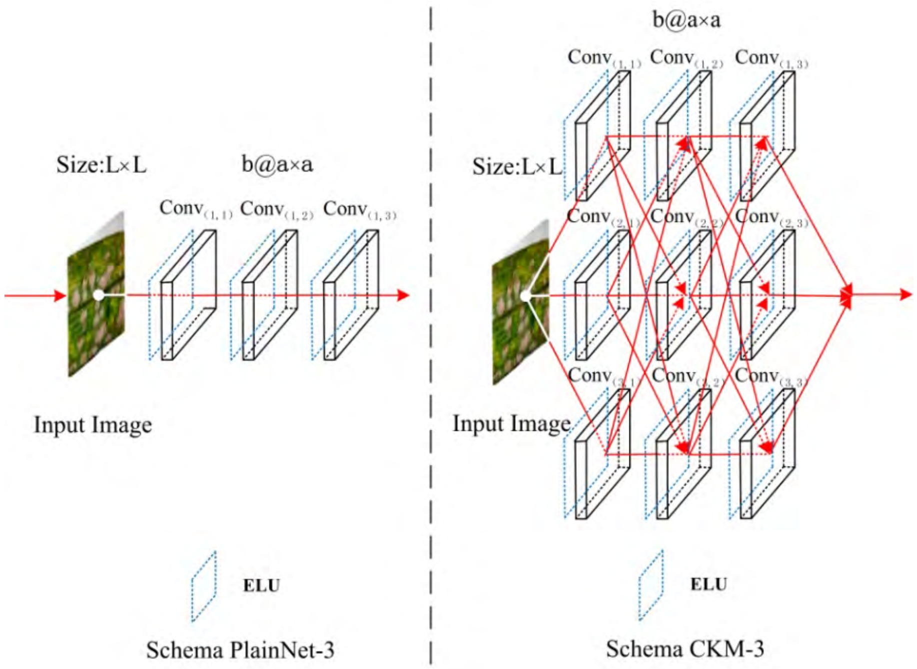


FIGURE 5. The structure of Schema PlainNet-3 and Schema CKM-3. Left: a 3-layer plain network as a reference. Right: a 3 × 3 convolutional kernel matrix.

TABLE 3. Data streams in Schema PlainNet-3 and Schema CKM-3.

Name	Data Stream
Schema PlainNet-3	$\text{Conv}_{(1,1)} \rightarrow \text{Conv}_{(1,2)} \rightarrow \text{Conv}_{(1,3)}$
Schema CKM-3	$\text{Conv}_{(1,1)} \rightarrow \text{Conv}_{(1,2)} \rightarrow \text{Conv}_{(1,3)}$, $\text{Conv}_{(1,1)} \rightarrow \text{Conv}_{(2,2)} \rightarrow \text{Conv}_{(1,3)}$, $\text{Conv}_{(1,1)} \rightarrow \text{Conv}_{(3,2)} \rightarrow \text{Conv}_{(1,3)}$,
	$\text{Conv}_{(1,1)} \rightarrow \text{Conv}_{(1,2)} \rightarrow \text{Conv}_{(2,3)}$, $\text{Conv}_{(1,1)} \rightarrow \text{Conv}_{(2,2)} \rightarrow \text{Conv}_{(2,3)}$, $\text{Conv}_{(1,1)} \rightarrow \text{Conv}_{(3,2)} \rightarrow \text{Conv}_{(2,3)}$,
	$\text{Conv}_{(1,1)} \rightarrow \text{Conv}_{(1,2)} \rightarrow \text{Conv}_{(3,3)}$, $\text{Conv}_{(1,1)} \rightarrow \text{Conv}_{(2,2)} \rightarrow \text{Conv}_{(3,3)}$, $\text{Conv}_{(1,1)} \rightarrow \text{Conv}_{(3,2)} \rightarrow \text{Conv}_{(3,3)}$,
	$\text{Conv}_{(2,1)} \rightarrow \text{Conv}_{(1,2)} \rightarrow \text{Conv}_{(1,3)}$, $\text{Conv}_{(2,1)} \rightarrow \text{Conv}_{(2,2)} \rightarrow \text{Conv}_{(1,3)}$, $\text{Conv}_{(2,1)} \rightarrow \text{Conv}_{(3,2)} \rightarrow \text{Conv}_{(1,3)}$,
	$\text{Conv}_{(2,1)} \rightarrow \text{Conv}_{(1,2)} \rightarrow \text{Conv}_{(2,3)}$, $\text{Conv}_{(2,1)} \rightarrow \text{Conv}_{(2,2)} \rightarrow \text{Conv}_{(2,3)}$, $\text{Conv}_{(2,1)} \rightarrow \text{Conv}_{(3,2)} \rightarrow \text{Conv}_{(2,3)}$,
	$\text{Conv}_{(2,1)} \rightarrow \text{Conv}_{(1,2)} \rightarrow \text{Conv}_{(3,3)}$, $\text{Conv}_{(2,1)} \rightarrow \text{Conv}_{(2,2)} \rightarrow \text{Conv}_{(3,3)}$, $\text{Conv}_{(2,1)} \rightarrow \text{Conv}_{(3,2)} \rightarrow \text{Conv}_{(3,3)}$
	$\text{Conv}_{(3,1)} \rightarrow \text{Conv}_{(1,2)} \rightarrow \text{Conv}_{(1,3)}$, $\text{Conv}_{(3,1)} \rightarrow \text{Conv}_{(2,2)} \rightarrow \text{Conv}_{(1,3)}$, $\text{Conv}_{(3,1)} \rightarrow \text{Conv}_{(3,2)} \rightarrow \text{Conv}_{(1,3)}$,
	$\text{Conv}_{(3,1)} \rightarrow \text{Conv}_{(1,2)} \rightarrow \text{Conv}_{(2,3)}$, $\text{Conv}_{(3,1)} \rightarrow \text{Conv}_{(2,2)} \rightarrow \text{Conv}_{(2,3)}$, $\text{Conv}_{(3,1)} \rightarrow \text{Conv}_{(3,2)} \rightarrow \text{Conv}_{(2,3)}$,
	$\text{Conv}_{(3,1)} \rightarrow \text{Conv}_{(1,2)} \rightarrow \text{Conv}_{(3,3)}$, $\text{Conv}_{(3,1)} \rightarrow \text{Conv}_{(2,2)} \rightarrow \text{Conv}_{(3,3)}$, $\text{Conv}_{(3,1)} \rightarrow \text{Conv}_{(3,2)} \rightarrow \text{Conv}_{(3,3)}$

The number of neurons, link channels, and training parameters of two schemas are shown in **Table** 4.

In **Table** 3 and **Table** 4, the numbers of data streams and link channels in Schema CKM-3 are both 27 times those of Schema PlainNet-3, which is a distinct improvement compared with Schema CKM-2. The number of neurons in Schema CKM-3 is three times that of Schema PlainNet-3. Moreover, the number of training parameters is also increased by three times. It can be seen that the improvement of neurons and link channels is more significant than Schema CKM-2, while the number of training parameters is within the acceptable range.

TABLE 4. The number of neurons, link channels and training parameters in Schema PlainNet-3 and Schema CKM-3.

Name	Number of neurons	Number of link channels	Number of training parameters
Schema PlainNet-3	$b[(L-a+1)^2+(L-2a+2)^2+(L-3a+3)^2]$	$(a^2+1)b[(L-a+1)^2+(L-2a+2)^2+(L-3a+3)^2]$	$3(a^2+1)b$
Schema CKM-3	$3b[(L-a+1)^2+(L-2a+2)^2+(L-3a+3)^2]$	$27(a^2+1)b[(L-a+1)^2+(L-2a+2)^2+(L-3a+3)^2]$	$9(a^2+1)b$

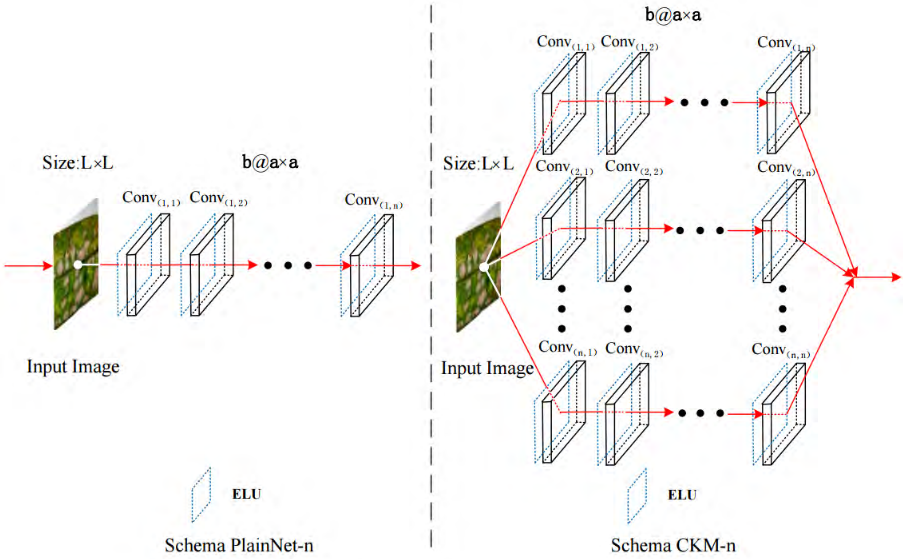


FIGURE 6. The structure of Schema PlainNet-n and Schema CKM-n. Left: an n-layer plain network as a reference. Right: an n × n convolutional kernel matrix.

3) SCHEMA PlainNet-N AND SCHEMA CKM-N

With the improvement of hardware, the implementation of convolutional kernel matrix with bigger size may be allowed, whose structure is like Schema CKM-n in **Fig.** 6. It is composed of n^2 convolutional layers ($\mathrm{Conv}_{(1,1)}$, $\mathrm{Conv}_{(2,1)}, \ldots, \mathrm{Conv}_{(n,1)}$; $\mathrm{Conv}_{(1,2)}, \mathrm{Conv}_{(2,2)}, \ldots, \mathrm{Conv}_{(n,2)}$; $\ldots$; $\mathrm{Conv}_{(1,n)}$, $\mathrm{Conv}_{(2,n)}, \ldots \mathrm{Conv}_{(n,n)}$) and each one owns b a × a convolution filters. Layers in adjacent columns are fully connected with each other, so there is n^n data streams in Schema CKM-n. As a reference, Schema PlainNet-n consists of n linearly sequenced convolutional layers ($\mathrm{Conv}_{(1,1)}$, $\mathrm{Conv}_{(1,2)}, \ldots \mathrm{Conv}_{(1,n)}$) and each one also has b a × a convolution filters. The data streams of Schema PlainNet-n and Schema CKM-n are shown in **Table** 5.

The number of neurons, link channels and training parameters in Schema PlainNet-n and Schema CKM-n are calculated by Eq. (1) to Eq. (6), where L, b, and a denote the input image size, the number and the size of convolution filters respectively, and they are initialized to 256, 10, and 3 respectively. $Num\_PlainNet - n_{Neu}$, $Num\_PlainNet - n_{lc}$, and $Num\_PlainNet - n_{tp}$ represent the number of neurons, link channels, and training parameters of Schema PlainNet-n while $Num\_CKM - n_{Neu}$, $Num\_CKM - n_{lc}$, and $Num\_CKM - n_{tp}$ represent those of Schema CKM-n, and n denotes the size of convolutional kernel matrix. The corresponding functions of each equation are illustrated in **Fig.** 7 and **Fig.** 8.

It can be seen in equations 5, 6, 8, and 9, that the numbers of neurons and link channels in Schema CKM-n are n and n^n times those of Schema PlainNet-n, respectively. This means sufficient convolution filters and non-linear mappings are available for better features extraction. From **Fig.** 7 (a), **Fig.** 7 (b), **Fig.** 8 (a), and **Fig.** 8 (b), we can also see that this improvement generated by convolutional kernel matrix becomes more significant with the increase in matrix size. Especially for link channels, the number of them in Schema CKM-n is seven orders of magnitude larger than that of Schema PlainNet-n, when the matrix size is greater than or equal to six. In terms of training parameters, from Eq. (7) and Eq. (10), we can see the number of it in Schema CKM-n is n times that of Schema PlainNet-n, but it is noteworthy that the link channels simultaneously increased by n^n times. Therefore, the matrix structure of convolutional layers is an effective and economical way to boost the representational ability of the model. Moreover, from **Fig.** 7 (c) and **Fig.** 8 (c) we observe that the number of training parameters in Schema CKM-n is within 7,000, when the matrix size is equal or less than eight, which is just one order of magnitude larger than that of Schema PlainNet-n. It is not enough to lead in unacceptable computational budget and the curse of dimensionality. Therefore, the obvious computational burden and serious accuracy loss will not occur at the experimental stage. Schema PlainNet-n and Schema

TABLE 5. Data streams in Schema PlainNet-n and Schema CKM-n.

Name	Data stream
Schema PlainNet-n	Conv$_{(1,1)}$ → Conv$_{(1,2)}$ → ... → Conv$_{(1,n)}$
Schema CKM-n	Conv$_{(1,1)}$ → Conv$_{(1,2)}$ → ... → Conv$_{(1,n)}$, Conv$_{(1,1)}$ → Conv$_{(1,2)}$ → ... → Conv$_{(2,n)}$, …, Conv$_{(1,1)}$ → Conv$_{(1,2)}$ → ... → Conv$_{(n,n)}$, Conv$_{(1,1)}$ → Conv$_{(1,2)}$ → ... → Conv$_{(2,n)}$, Conv$_{(1,1)}$ → Conv$_{(1,2)}$ → ... → Conv$_{(1,n)}$, …, Conv$_{(1,1)}$ → Conv$_{(1,2)}$ → ... → Conv$_{(1,n)}$ . . . Conv$_{(n,1)}$ → Conv$_{(n,2)}$ → ... → Conv$_{(1,n)}$, Conv$_{(n,1)}$ → Conv$_{(n,2)}$ → ... → Conv$_{(n,2)}$, …, Conv$_{(n,1)}$ → Conv$_{(n,2)}$ → ... → Conv$_{(n,n)}$

CKM-n are represented in (5)–(7) and (8)–(10), respectively, as shown at the bottom of this page.

The time complexity of one convolutional kernel matrix can be calculated by Eq. (11):

$$\begin{cases} \text{Time} \sim O(\sum_{i=1}^{N} NM_i^2 \cdot NK_i^2 \cdot N^N C_{i-1} \cdot N^N C_i) \\ M = (X - K + 2 * Padding)/Stride + 1 \\ C_i = (K^2 + 1)F(X - iK + 1)^2 \\ N \in \{x | x \geq 2, x \in Z\} \end{cases} \tag{11}$$

where N denotes the matrix size, M and X denote the size of output and input feature maps respectively, K and F denote the size and number of convolution filters, C_{i-1} and C_i denote the channels of (*i-1*)th and *i*th layer respectively. From the Eq. (11) we can perceive that the amount of neurons and channels are related with time complexity, and so is the amount of parameters. If the neurons, link channels, and parameters all grow sharply, it will undoubtedly result in intolerable time complexity (often occurs in deep plain networks [72], [74], [79] with many linearly stacked layers). So the convolutional kernel matrix was proposed to increase the neurons and link channels whilst restraining parameters growth, and the previous two are beneficial for representational ability enhancing. Although the neurons and link channels grow rapidly, the total time complexity is not so obvious for the suppression of parameter growth. Moreover, thanks to the activation function "ELU" (see section II, A, 2) and optimization strategy "SGD + momentum" (see section IV, B, 4), the convergence rate has been accelerated to some extent. From the training phase (see section IV, B, 5) we can observe that only ten more epochs (about four more hours) are required to achieve convergence. The time complexity can also be diluted by integrated mature tricks. It also demonstrates that the additional time complexity is not obvious.

In conclusion, the above three comparisons (see section 1, 2, and 3) demonstrate that convolutional kernel matrix provided significantly better performance than the plain networks. This proves that the matrix structure helps with achieving a substantial increase of data streams, neurons, and link channels at a tolerable increase of computational requirements for affordable parameters addition. This way, the curse of dimensionality will not appear within a proper matrix size.

$$\boxed{\text{Schema PlainNet-n}} \left\{ \begin{aligned} Num\_PlainNet - n_{Neu} &= b[(L-a+1)^2 + (L-2a+2)^2 + \ldots + (L-na+n)^2] \\ &= b\sum_{i=1}^{n}\left[(L-ia+i)^2\right], \quad n \in \{x|x \geq 2, x \in Z\} \qquad (5) \\ Num\_PlainNet - n_{lc} &= (a^2+1)b[(L-a+1)^2 + (L-2a+2)^2 + \ldots + (L-na+n)^2] \\ &= (a^2+1)b\sum_{i=1}^{n}[(L-ia+i)^2], \quad n \in \{x|x \geq 2, x \in Z\} \qquad (6) \\ Num\_PlainNet - n_{tp} &= n(a^2+1)b, \quad n \in \{x|x \geq 2, x \in Z\} \qquad (7) \end{aligned} \right.$$

$$\boxed{\text{Schema CKM-n}} \left\{ \begin{aligned} Num\_CKM - n_{Neu} &= nb[(L-a+1)^2 + (L-2a+2)^2 + \ldots + (L-na+n)^2] \\ &= nb\sum_{i=1}^{n}\left[(L-ia+i)^2\right], \quad n \in \{x|x \geq 2, x \in Z\} \qquad (8) \\ Num\_CKM - n_{lc} &= n^n(a^2+1)b[(L-a+1)^2 + (L-2a+2)^2 + \ldots + (L-na+n)^2] \\ &= n^n(a^2+1)b\sum_{i=1}^{n}[(L-ia+i)^2], \quad n \in \{x|x \geq 2, x \in Z\} \qquad (9) \\ Num\_CKM - n_{tp} &= n^2(a^2+1)b, \quad n \in \{x|x \geq 2, x \in Z\} \qquad (10) \end{aligned} \right.$$

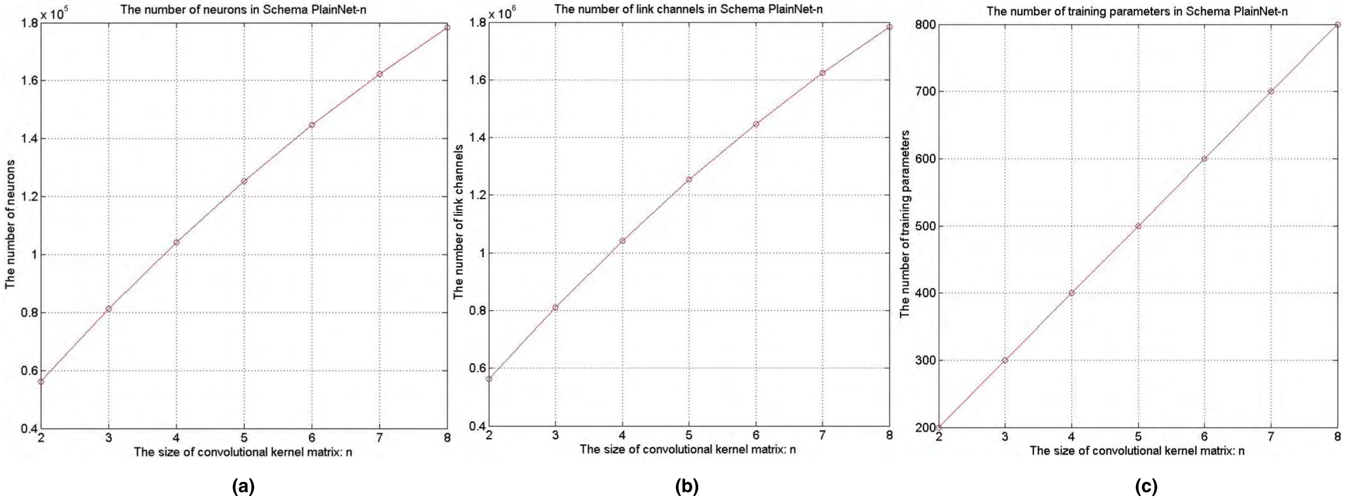


FIGURE 7. The number of neurons, link channels, and training parameters in Schema PlainNet-n. (a) The number of neurons in Schema PlainNet-n. (b) The number of link channels in Schema PlainNet-n. (c) The number of training parameters in Schema PlainNet-n.

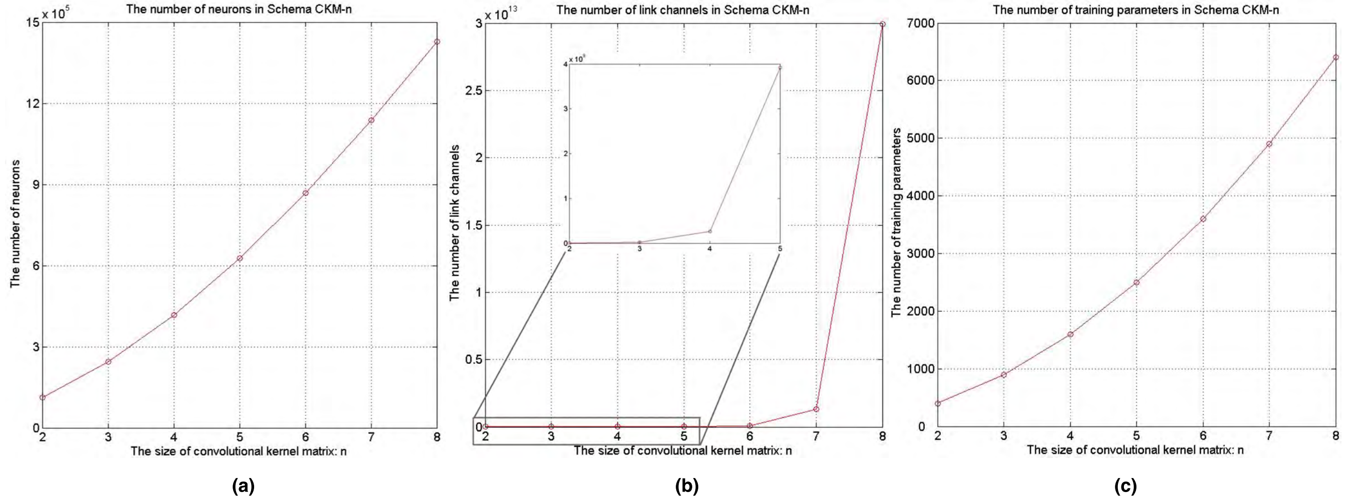


FIGURE 8. The number of neurons, link channels, and training parameters in Schema CKM-n. (a) The number of neurons in Schema CKM-n. (b) The number of link channels in Schema CKM-n. (c) The number of training parameters in Schema CKM-n.

Sharing the above advantages, M-bCNN could easily relish accuracy gains from considerably increased depth, producing efforts substantially better than plain networks.

III. DATA DESCRIPTION

In this work, winter wheat leaf diseases images were utilized as the experimental samples of fine-grained classification for their strong similarity with subordinate categories in some cases. At present, no large-scale image set of wheat leaf diseases is publicly available. So 16,652 high-fidelity images were collected from several wheat planting areas of Shandong province and were assigned as the original image set. Then an augmented database containing 83,260 images was constructed by five augmentation methods. The original and augmented image sets were utilized as the training and testing samples, respectively. To the best of our knowledge, this is the first available large-scale high-resolution images sets of winter wheat leaf diseases.

A. IMAGE ACQUISITION

From the wheat planting bases of Shandong Province, China, 16,652 winter wheat leaf images containing eight categories were collected from the field using Canon EOS 80D camera. They were acquired between 8:00 a.m. and 5:00 p.m., and the distance of the camera from the leaf was three to seven cm. The image format was JPEG and each one was a 24-bit color bitmap. Each image included only one disease and was classified into one corresponding ground truth category by plant protection experts. The original image set was utilized as testing sample. See **Table** 6 for randomly selected samples.

From **Table** 6 we can observe that the images of some subcategories, within one common basic-level category, have strong similarities. For example, the images of Powdery Mildew and Cochliobolus Heterostrophus are similar to each other, and they belong to the same basic category of Spore Parasitism. Therefore, it is generally much harder to classify each image with in its true category and this is a meaningful

TABLE 6. Samples of winter wheat leaf disease images.

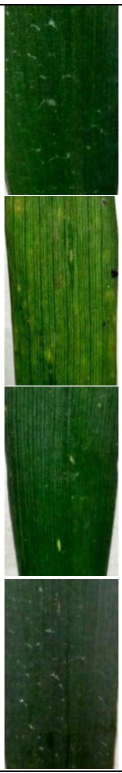
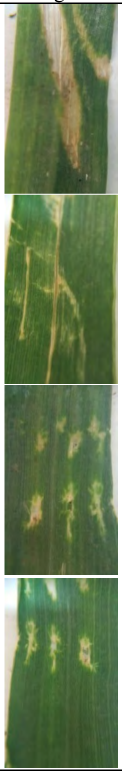
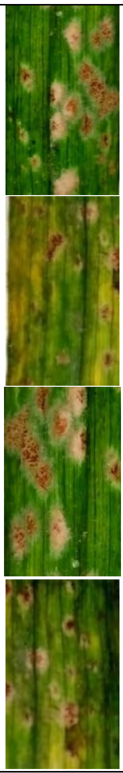
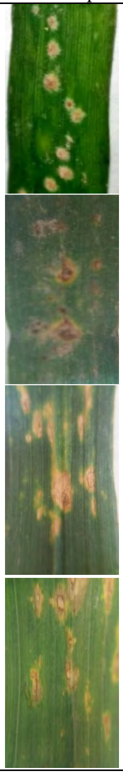
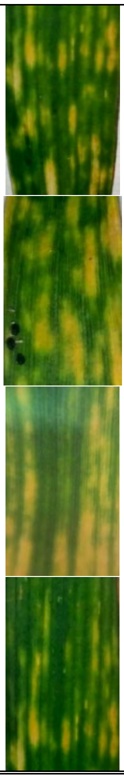
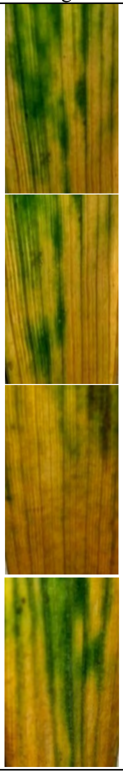
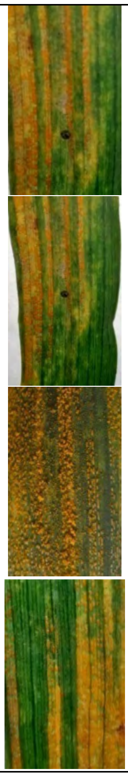
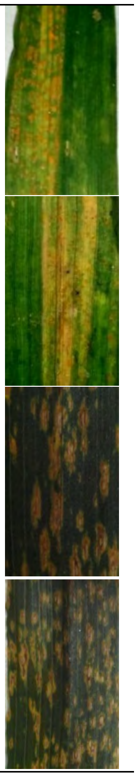
Basic-category			Spore Parasitism		Bacterial Disease		Rust Disease	
Sub-category	Normal Leaf	Mechanical Damage Leaf	Powdery Mildew	Cochliobolus Heterostrophus	Bacterial Leaf Streak	Bacterial Leaf Blight	Stripe Rust	Leaf Rust
Samples								

TABLE 7. Number and proportion of each category in original image set.

Category	Normal Leaf	Mechanical Damage Leaf	Powdery Mildew	Bacterial Leaf Streak	Cochliobolus Heterostrophus	Stripe Rust	Leaf Rust	Bacterial Leaf Blight
Number	2,032	2,472	2,364	1,924	2,092	1,878	2,122	1,766
Proportion /%	0.122	0.148	0.141	0.115	0.125	0.112	0.127	0.110

dataset for model evaluation of fine-grained classification. The number and proportion of each category is shown in **Table** 7.

B. IMAGE AUGMENTATION

Adequately labeled samples can reduce under-fitting in the model training process [83]. In order to increase the number and diversity of original images and enable extracted features to own the robustness of rotation, translation, and scaling, etc., an augmented image set was constructed through noise addition [84], color jittering [68], PCA jittering [68], rotation blur [55], and scaling blur [85] for their implementation simplicity and satisfactory performances proved in previous researches [23], [52], [72], [73]. These methods simulate the interferences of noise, illumination fluctuation, and object jitter which are frequently encountered during the acquisition process and practical application scenarios. See **Table** 8 for details and **Table** 9 for the processed images corresponding to each method.

Finally, we augmented dataset of 83,260 images that were enough for the model's convergence. This image set was utilized as the training sample and the number and proportion of each category is shown in **Table** 10.

IV. EXPERIMENTS

A. EXPERIMENTAL ENVIRONMENTS

Training a deep CNN through a large number of iterations largely relies on high-performance graphics processing units (GPUs). We run the experiments using multiple GPUs on NVIDIA (R) GeForce GTX 1080 graphics card. Its basic configuration is shown in **Table** 11.

The computer was a HP EliteDesk 880 G2 TWR with an Intel(R) Core i7 6700K (3.40 GHz) processor and 16 GB memory. The operating system was Ubuntu 16.04.1 (64 bits). Implementation of the M-bCNN used TensorFlow, an open-source machine learning framework for numerical computation developed by Google Brain Team. The Python was

TABLE 8. Methods of Image augmentation used in the study.

Method	Details
Noise Addition [84]	1) A 30% Gauss noise is added to the original image, with an offset of 0.2 and a standard deviation of 0.3.
Color Jittering [68]	1) The hue, saturation and brightness are increased by 20%, 2) The contrast is increased by 30%, 3) The sharpness is decreased by 10%.
PCA Jittering [68]	1) The means and standard deviations of R, G, and B channels are calculated, and the input signals are normalized. 2) The dimension of image pixel $I_{xy} = [I_{Rxy}, I_{Gxy}, I_{Bxy}]^{\mathrm{T}}$ is reduced through PCA, and three principal direction vectors p_1 , p_2 , p_3 , and three features λ_1 , λ_2 , λ_3 are obtained. 3) Add $[p_1, p_2, p_3][\alpha_1\lambda_1, \alpha_2\lambda_2, \alpha_3\lambda_3]^{\mathrm{T}}$ to $I_{xy} = [I_{Rxy}, I_{Gxy}, I_{Bxy}]^{\mathrm{T}}$. α_i is a variable satisfying the mean of 0 and the difference of 0.1.
Rotation Blur [55]	1) The original image is radially blurred by 10 rotating fuzzy units.
Scaling Blur [85]	1) The original image is radially blurred by 30 scaling fuzzy units.

TABLE 9. Augmented images as a consequence of five individual augmentation methods.

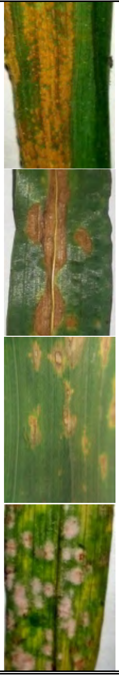
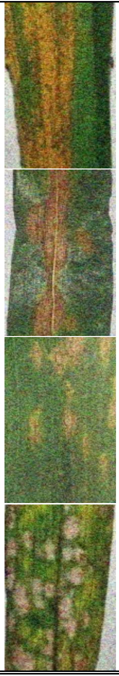
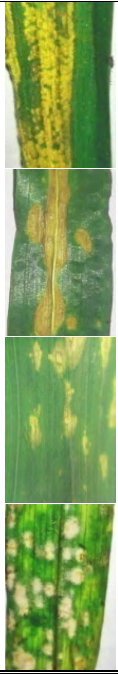
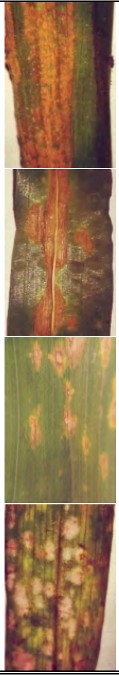
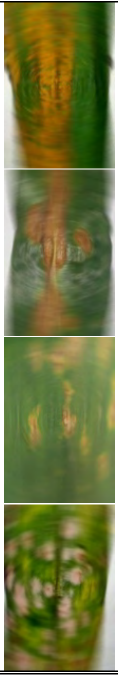
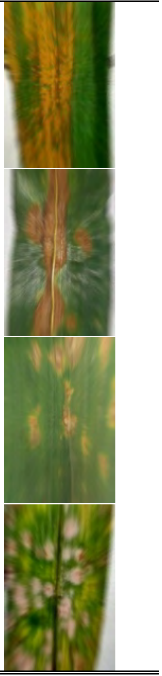
Name	Normal Image	Noise Addition [84]	Color Jittering [68]	PCA Jittering [68]	Rotation Blur [55]	Scaling Blur [85]
Sample						

TABLE 10. Number and proportion of each category in augmented image set.

Category	Normal Leaf	Mechanical Damage Leaf	Powdery Mildew	Bacterial Leaf Streak	Cochliobolus Heterostrophus	Stripe Rust	Leaf Rust	Bacterial Leaf Blight
Number	10,160	12,370	11,820	9,620	10,460	9,390	10,610	8,830
Proportion /%	0.122	0.148	0.141	0.115	0.125	0.112	0.127	0.110

utilized as the programming language to adapt to the core of TensorFlow.

B. EVALUATION METRIC

The following metrics are considered to evaluate the model. First and foremost, the accuracy is widely implemented for the target classification and recognition. However, the performance of the model cannot be illustrated sufficiently only with accuracy. So, we use confusion matrix, also known as error matrix in supervised learning, which clearly depicts the actual and predicted categories in each column and row respectively. Moreover, the precision, recall, and F1-scores across individual categories are utilized to evaluate the performances of the classifier model directly.

C. MODEL TRAINING

Before the training starts, 70% of the images in each category are randomly selected as training samples and the

TABLE 11. Basic Characteristics of GPUs.

Configuration Parameter	Parameter Value
Chip model	NVIDIA GeForce GTX 1080
RAM capacity	8,192M
Core frequency	1,759/1,936MHz
Memory frequency	10,206/10,400MHz
Stream processor	2,560
Raster processing unit	64
RAMDAC frequency	400MHz
Maximum resolution	7,680 × 4,320

remaining 30% are utilized as validation samples. Since the augmented dataset is already balanced, this sampling method can ensure the inter-class balance.

1) OPTIMIZATION OBJECTIVE

Suppose there are N training samples and the feature vector of nth ($1 \leq n \leq N$) sample is denoted as $x^n = (x_1^n, x_2^n, \ldots, x_m^n)$, where m represents the number of dimensions. The corresponding actual output vector and the expected output vector are $y^n = (y_1^n, y_2^n, \ldots, y_m^n)$ and $EO^n = (EO_1^n, EO_2^n, \ldots, EO_c^n)$, where c denotes the numbers of output vectors. Then the optimization objective of M-bCNN is the mean squared error of all samples, as shown below:

$$E^N = \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{c}(y_m^n - EO_m^n)^2 \tag{12}$$

2) LOSS FUNCTION

The standard cross-entropy [86] is utilized as the loss function during the model training stage and it is defined as in Eq. (13):

$$E = -\frac{1}{n}\sum_{x}[y \ln y' + (1-y)\ln(1-y\text{‘})] \tag{13}$$

where y and y' denote the expected and actual output, respectively.

3) REGULARIZATION TERM

In order to better resist over-fitting and vanishing gradient, L2 regularization is exploited and is shown in Eq. (14):

$$L_2 = \frac{1}{2n}\lambda\sum_{w_i} w_i^2 \tag{14}$$

where w and n denote model parameters and the number of samples respectively, λ is the weight decay and is assigned to 0.001.

4) OPTIMIZATION STRATEGY

In pursuit of faster training speed, the strategy of "SGD + momentum" is utilized as the optimization algorithm. Its optimization speed is $1/1 - \alpha$ times faster than that of SGD [87], where α denotes momentum and ranges $0 < \alpha < 1$. The optimization process of "SGD+momentum" is shown in Eq. (15):

$$\begin{cases} J(\theta) = \dfrac{1}{N}\displaystyle\sum_{i=1}^{N}\dfrac{1}{2}(y^i - h_\theta(x^i))^2 \\ v = \alpha v - \varepsilon J(\theta) \\ \theta = \theta + v \\ N = 1, 2, 3, \ldots, n \end{cases} \tag{15}$$

where x^i and y^i denote input and output signals, $J(\theta)$ and $h(\theta)$ is the gradient estimation and fitting function, θ is the parameter needed to be optimized and it decides $h(\theta)$, ε is the learning rate and is initialized into 0.01, its decay steps and decay rate are assigned to 3,000 and 0.1 respectively, which means that ε is divided by ten at every 3,000 iterations, momentum α is assigned to 0.99 for 100 times improvement of optimization speed, and v is the learning speed that is refreshed after every iteration.

Finally, Batch normalization (BN) [88] is adopted right after each convolution layer and all models are trained from scratch.

5) TRAINING IMPLEMENTATION

The structures of Schemas CKM-2 and CKM-3 are realized in models M-bCNN-CKM-2 and M-bCNN-CKM-3 respectively, and contrasted with two representative plain networks, AlexNet [68] and VGG-16 [73], for comparison studies. In the same experimental environment, M-bCNN-CKM-2 and M-bCNN-CKM-3 are first pre-trained on the ImageNet dataset [69] for their large parameters, and then four models are fine-tuned for up to 100 epochs end-to-end by SGD + momentum with back-propagation on the augmented image set, where the mini-batch size is 50. **Fig.** 9 (a), (b), and (c) depict the accuracy of training and validation image set throughout the whole procedure.

Fig. 9 (a)-(c) compare the training and validation accuracy of four models. **Fig.** 9 (a) shows that M-bCNN-CKM-2 and M-bCNN-CKM-3 converged after about 50 training epochs. The results indicate that the two models have equivalent accuracy for the training image set, whereas for validation image set, the validation accuracy of M-bCNN-CKM-3 is better than that of M-bCNN-CKM-2. Based on these result, the M-bCNN-CKM-2 was then compared with AlexNet and VGG-16 as shown in **Fig.** 9 (b) and (c). The results demonstrate that the training and validation accuracy of M-bCNN-CKM-2 are both higher than those of AlexNet and VGG-16, and only ten more training epochs are required to achieve convergence.

According to the results in **Fig.** 9 (a)-(c), the model that maximized the accuracy for the validation image set is considered to be the best. **Table** 12 shows the training accuracy, validation accuracy, training epoch, and training time for each model. M-bCNN-CKM-3, which achieved the highest validation accuracy, is the best performing model. When the models are convergent, the highest validation accuracies of M-bCNN-CKM-2 and M-bCNN-CKM-3 are about 91.32% and 96.5% respectively, which are

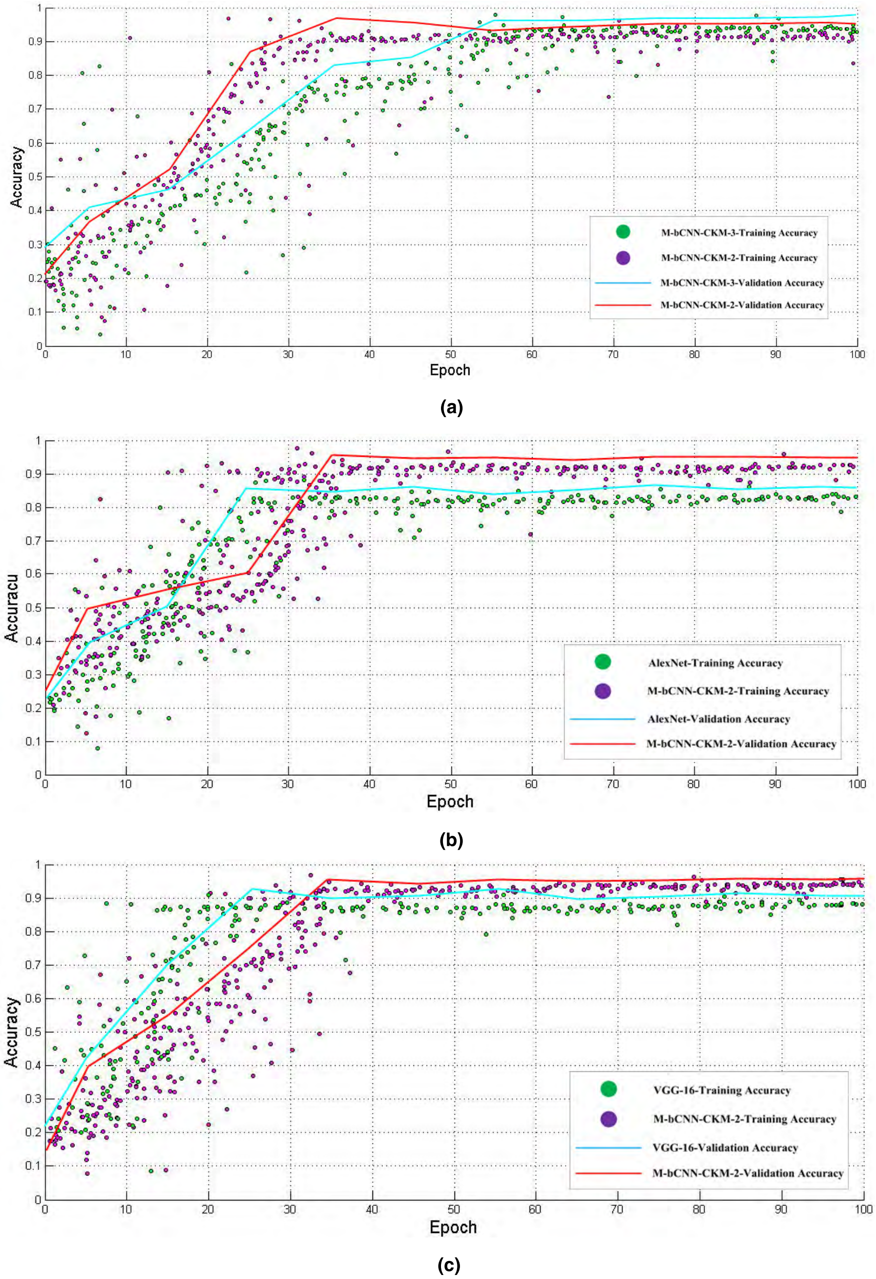


FIGURE 9. Accuracy in the training image set and validation image set. (a) Iteration of training accuracies changes and validation accuracies changes by M-bCNN-CKM-2 and M-bCNN-CKM-3. (b) Iteration of training accuracies changes and validation accuracies changes by M-bCNN-CKM-2 and AlexNet. (c) Iteration of training accuracies changes and validation accuracies changes by M-bCNN-CKM-2 and VGG-16.

obviously higher than those of AlexNet and VGG-16 (83.12% and 88.54% respectively). M-bCNN-CKM-2 and M-bCNN-CKM-3 achieved higher validation accuracies of fine-grained classification for wheat leaf diseases' images, but required just about four more hours to converge. It suggests that the convolutional neural network is effective both in boosting up the representational ability and suppressing parameter growth, while the training and validation accuracies do not suffer the penalty of the curse of dimensionality.

D. FEATURE VISUALIZATION

In order to get a clearer understanding of how and why the models work, guided-backpropagation and deconvolution [89] are both utilized to visualize the constantly updating filters of the model throughout the whole training stage. In the course of the experimental iterations, the visualization of some randomly selected filters in M-bCNN-CKM-3 is shown in **Fig.** 10.

We can view the above filters as the learned feature descriptors encoding the distinctive fusion structures. It is

TABLE 12. Accuracy and epoch of the best models, and training time (h).

Model	M-bCNN-CKM-2	M-bCNN-CKM-3	AlexNet [68]	VGG-16 [73]
Training Accuracy(%)	0.916	**0.942**	0.827	0.871
Validation Accuracy(%)	0.913	**0.965**	0.831	0.885
Training Epoch	33	**48**	28	32
Training Time (h)	11.17	**13.03**	8.36	9.09

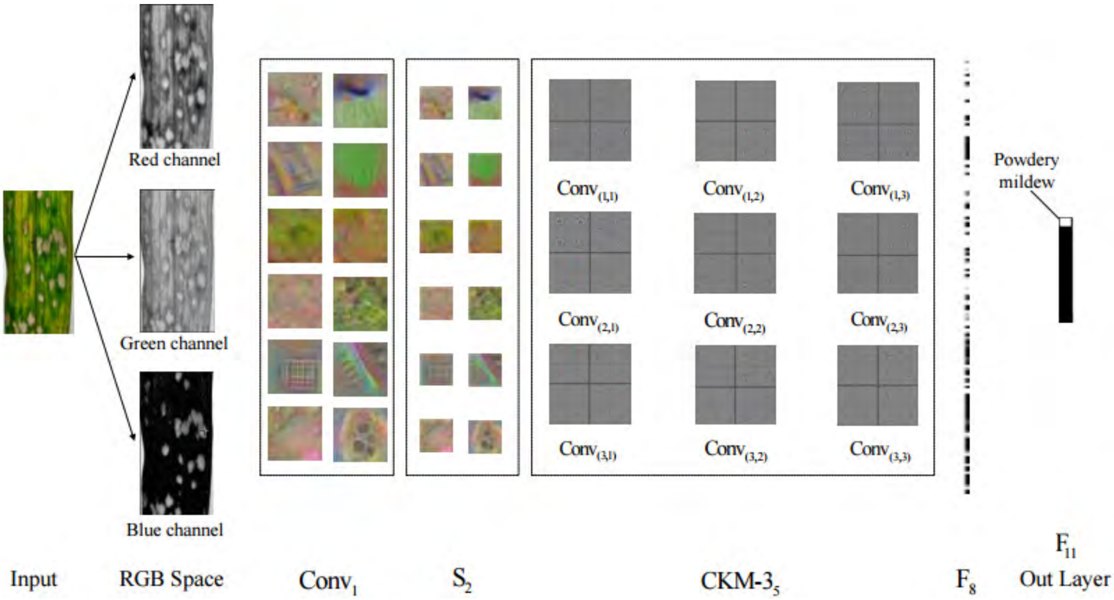


FIGURE 10. The filters of some hidden layers in M-bCNN-CKM-3 visualized as small patches.

noticeable that despite each filter is independent in $Conv_1$ and S_1, our filters in $CKM\text{-}3_5$ smoothly change during training stage. In this manner, they provide much richer and more meaningful domain-specific representations. In a scene, this also further demonstrates that using convolutional kernel matrix is a crucial requirement for a model to learn better representations.

E. MODEL TESTING

This section illustrates the performance verification of models in fine-grained image classification experiments. We utilized 16,652 original images as the testing samples. Then compared trained M-bCNN-CKM-2, M-bCNN-CKM-3 against AlexNet and VGG-16 and evaluated them in terms of individual accuracy, precision, recall, F1-socre, and overall accuracy.

In **Fig.** 11, four confusion matrixes, which compared the true category (Ordinate) against the predicted category (Abscissa), were calculated to describe the individual classification rate of each model. Note the color distribution of the confusion matrix and that the Normal Leaf and Bacterial Leaf Blight have high average classification rates (91.0% and 92.3% respectively) for all models. The large green area of healthy leaves and the golden appearance of leaves infected with Bacterial Leaf Blight make them easier to be distinguished with other sub-categories sharing one common basic-level category. Furthermore, M-bCNN-CKM-3 and AlexNet achieved the highest (89.6%) and the lowest (80.5%) average classification rate of eight categories, as obvious from **Fig.** 11 (b) and (c).

Fig. 12 and **Table** 13 depict the precision, recall, F1-score and accuracy of eight categories by four models for the testing image set. In **Fig.** 12, we observe that the precision (93.32%) of *Bacterial Leaf Blight* is the highest, while the recall (91.68%) of *Leaf Rust* is the highest. Other sub-categories sharing one basic-category are harder to be distinguished because of their strong similarity. The average precision (90.15%) and recall (88.62%) of M-bCNN-CKM-3 are the highest among these models, and those of AlexNet are the lowest (69.83% precision and 64.71% recall). In **Table** 13, the average F1-score (85.5%) and accuracy (90.1%) of M-bCNN-CKM-3 are also the highest for four models, and those of AlexNet are the lowest (51.75% F1-score and 72% accuracy). The F1-score and accuracy of these models indicate that M-bCNN-CKM-2 and M-bCNN-CKM-3 provide better performances than the other two plain networks in fine-grained image classification, and M-bCNN-CKM-3 is the top-performing approach compared to M-bCNN-CKM-2. Based on testing assessment, the convolutional kernel matrix is meaningful for boosting up representational ability compared with the linearly stacked layers, and the accuracy penalty caused by the curse of dimensionality has not appeared. Additionally, from the comparison of M-bCNN-CKM-2 and M-bCNN-CKM-3 (see also **Fig.** 9 (a) and

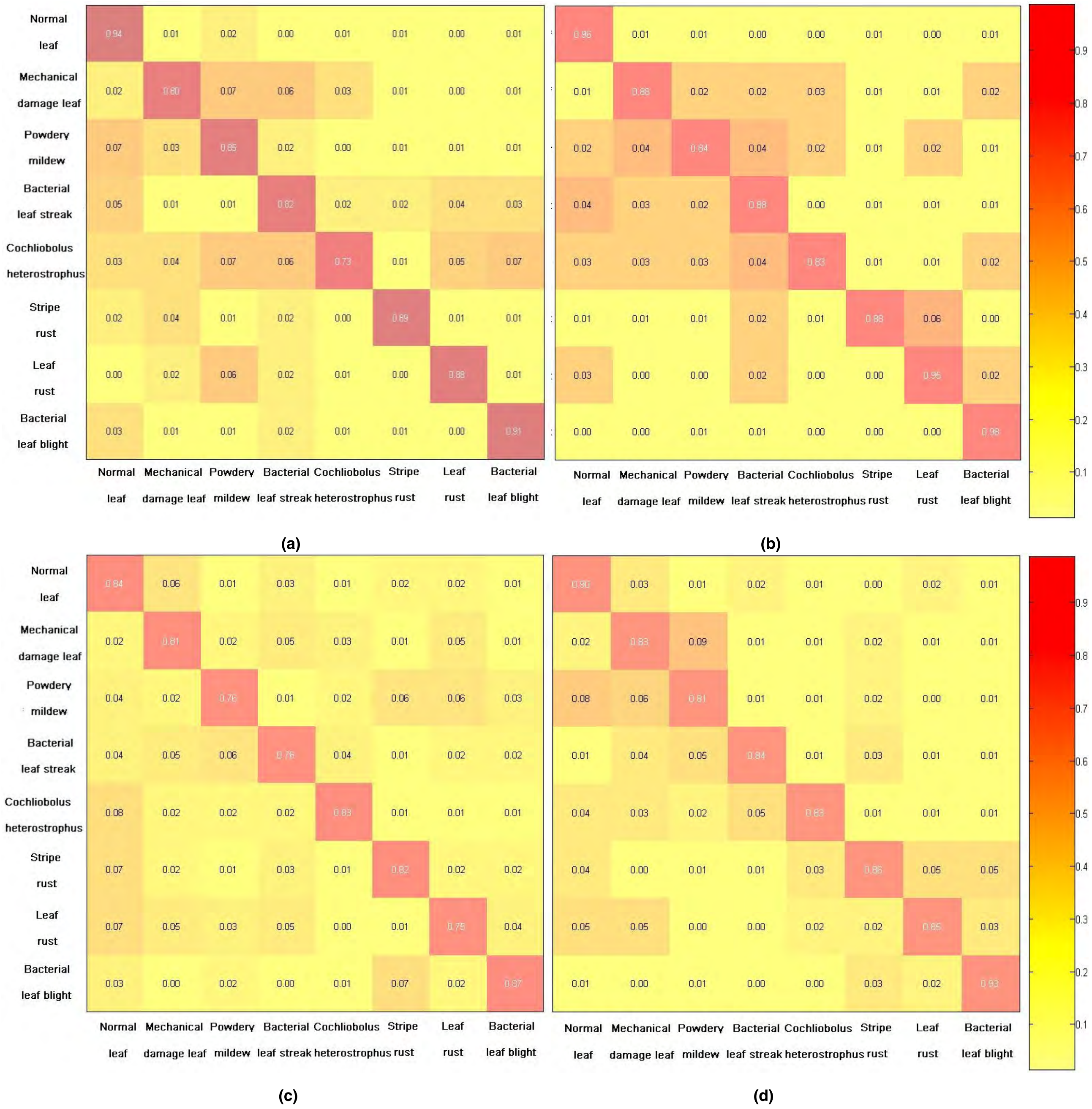


FIGURE 11. Confusion matrix of the testing results. (a) Illustration of individual classification rate by M-bCNN-CKM-2. (b) Illustration of individual classification rate by M-bCNN-CKM-3. (c) Illustration of individual classification rate by AlexNet. (d) Illustration of individual classification rate by VGG-16.

Table 12), we observe that this advantage becomes more significant as the matrix size increases. Sharing the advantages of convolutional kernel matrix, M-bCNN can easily acquire accuracy gains from the increased layers depth in the form of a matrix.

V. DISCUSSION

Recently, a number of studies have been conducted on fine-grained classification methods, and most of them provide promising performance in certain fields. Inspired by the design conceptions of parallel networks (e.g., Part-based CNN [8], Two-level Attention CNN [16], MCNN [55], GoogLeNet [72], ResNet [74], and Hypercolumn CNN [90]), we proposed a novel hybrid CNN structure codenamed M-bCNN, which leverages convolutional kernel matrixes to effectively increase the data streams, neurons, and link channels. The matrix-based architecture played an important role and the expected accuracy gains from it were delivered in the fine-grained image classification of wheat leaf diseases. The model's satisfying performance surpassed the two representative plain networks, i.e. AlexNet [68] and VGG-16 [73]. The experimental results and conclusions are

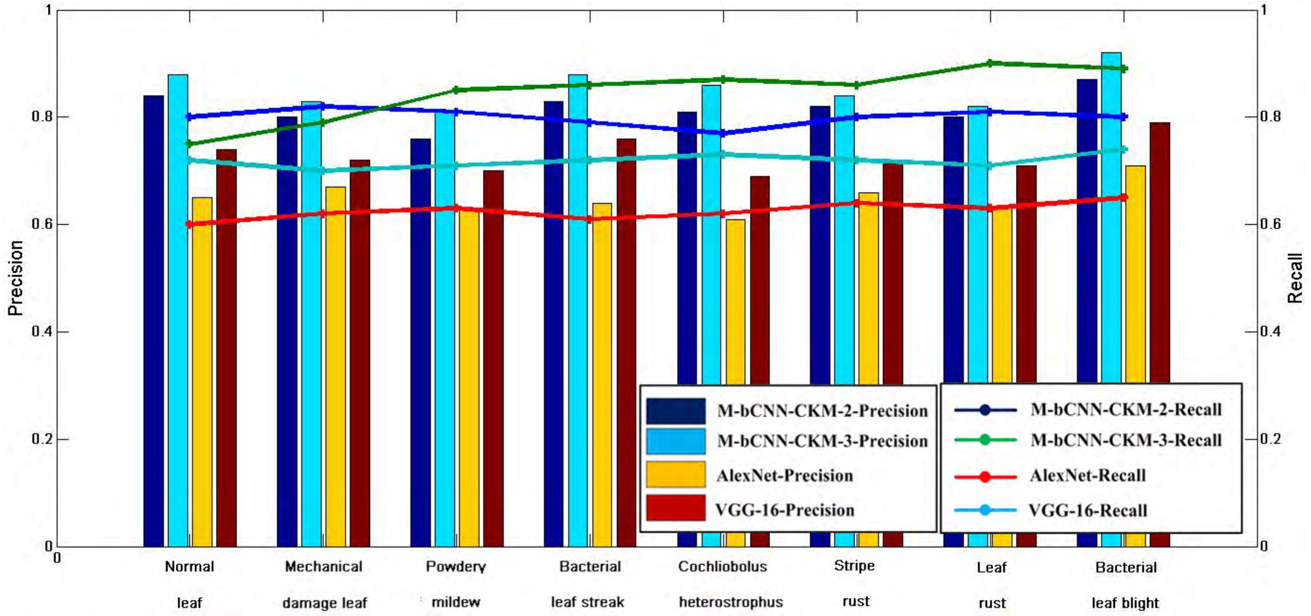


FIGURE 12. Precision and Recall.

TABLE 13. F1-score and accuracy.

	Normal Leaf	Mechanical Damage Leaf	Powdery Mildew	Bacterial Leaf Streak	Cochliobolus Heterostrophus	Stripe Rust	Leaf Rust	Bacterial Leaf Blight	Accuracy (%)
	F1-score								
M-bCNN-CKM-2(%)	0.72	0.70	0.72	0.73	0.73	0.75	0.73	0.76	0.83
M-bCNN-CKM-3(%)	**0.84**	**0.86**	**0.82**	**0.84**	**0.84**	**0.86**	**0.88**	**0.90**	**0.91**
AlexNet [68] (%)	0.52	0.53	0.51	0.50	0.52	0.51	0.52	0.53	0.72
VGG-16 [73] (%)	0.66	0.67	0.66	0.69	0.7	0.69	0.71	0.72	0.78

basically consistent with other studies on hierarchical models [8], [16], [55], [72], [74]. Parallelization and grading of neural networks is one of the developmental trends for deep learning. Our strategy might prove meaningful for other fine-grained tasks such as action or attribute categorization.

There could be one potential issue with the proposed model concerning the training phase: the convolutional kernel matrix may introduce heavy computations, when the iterations, training samples and matrix size are very large. Computations burden penalty is a common phenomenon that often occurs in the parallel networks (e.g., GoogLeNet [72], Two-level Attention CNN [16], and Hypercolumn CNN [90]), which cautiously sacrifice the proper algorithm efficiency for accuracy benefits. Consequently, it is important to carry out research on how to find an effective pruning mechanism for model compression, so that limited computational resources can be allocated scientifically and rationally. There is still room and potential to improve the performance to be at par with or even surpass the M-bCNN. One direction of future work is to delve deeper into the architecture optimization and integrate modified pruning mechanism into complex representational framework.

VI. CONCLUSION

In this work, we have proposed a unified CNN model, denoted M-bCNN, based on convolutional kernel matrix, for fine-grained image classification. The main advantage of convolutional kernel matrix is significant gains of data streams, neurons, and link channels at a modest increase of computational requirements compared to plain networks. We described the methodology of our architecture and positive effort on both representational ability enhancing and parameter growth inhibition.

The experiments demonstrated that the promising performances of our model compete against AlexNet and VGG-16 in the fine-grained image classification of wheat leaf diseases. Our approach yields solid evidence that convolutional kernel matrix is a feasible and useful idea in general, which provides a new path for the identification of crop diseases.

Future work directions are of two aspects: First, we will focus on optimizing the architecture and hyper-parameters of M-bCNN for other challenging fine-grained classification tasks. Second, we will try other models such as generative adversarial networks (GANs), regions with CNN (RCNN) to deal with semantic segmentation, object detection, and open-set recognition.

REFERENCES

[1] W. Huang *et al.*, "New optimized spectral indices for identifying and monitoring winter wheat diseases," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2516–2524, Jun. 2014, doi: 10.1109/JSTARS.2013.2294961.

[2] X. Zhang, Y. Qiao, F. Meng, C. Fan, and M. Zhang, "Identification of maize leaf diseases using improved deep convolutional neural networks," *IEEE Access*, vol. 6, pp. 30370–30377, 2018, doi: 10.1109/ACCESS.2018.2844405.

[3] K. R. Aravind, P. Raja, K. V. Mukesh, R. Aniirudh, R. Ashiwin, and C. Szczepanski, "Disease classification in maize crop using bag of features and multiclass support vector machine," in *Proc. 2nd Int. Conf. Inventive Syst. Control (ICISC)*, Coimbatore, India, Jan. 2018, pp. 1191–1196, doi: 10.1109/ICISC.2018.8398993.

[4] L. Han, M. S. Haleem, and M. Taylor, "A novel computer vision-based approach to automatic detection and severity assessment of crop diseases," in *Proc. Sci. Inf. Conf. (SAI)*, London, U.K., Jul. 2015, pp. 638–644, doi: 10.1109/SAI.2015.7237209.

[5] S. S. Chouhan, A. Kaul, U. P. Singh, and S. Jain, "Bacterial foraging optimization based radial basis function neural network (BRBFNN) for identification and classification of plant leaf diseases: An automatic approach towards plant pathology," *IEEE Access*, vol. 6, pp. 8852–8863, 2018, doi: 10.1109/ACCESS.2018.2800685.

[6] Y. Peng, X. He, and J. Zhao, "Object-part attention model for fine-grained image classification," *IEEE Trans. Image Process.*, vol. 27, no. 3, pp. 1487–1500, Mar. 2018.

[7] Y. Yu, Q. Jin, and C. W. Chen, "FF-CMnet: A CNN-based model for fine-grained classification of car models based on feature fusion," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, San Diego, CA, USA, Jul. 2018, pp. 1–6, doi: 10.1109/ICME.2018.8486443.

[8] M. Biglari, A. Soleimani, and H. Hassanpour, "A cascaded part-based system for fine-grained vehicle classification," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 273–283, Jan. 2018, doi: 10.1109/TITS.2017.2749961.

[9] J. Fang, Y. Zhou, Y. Yu, and S. Du, "Fine-grained vehicle model recognition using a coarse-to-fine convolutional neural network architecture," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 7, pp. 1782–1792, Jul. 2017, doi: 10.1109/TITS.2016.2620495.

[10] Q. Zhang, L. Zhuo, X. Hu, and J. Zhang, "Fine-grained vehicle recognition using hierarchical fine-tuning strategy for Urban surveillance videos," in *Proc. Int. Conf. Prog. Inform. Comput. (PIC)*, Shanghai, China, Dec. 2016, pp. 233–236, doi: 10.1109/PIC.2016.7949501.

[11] G. Sumbul, R. G. Cinbis, and S. Aksoy, "Fine-grained object recognition and zero-shot learning in remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 770–779, Feb. 2018, doi: 10.1109/TGRS.2017.2754648.

[12] Z. Ge, C. McCool, C. Sanderson, A. Bewley, Z. Chen, and P. Corke, "Fine-grained bird species recognition via hierarchical subset learning," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Quebec City, QC, Canada, Sep. 2015, pp. 561–565, doi: 10.1109/ICIP.2015.7350861.

[13] T. Saito, A. Kanezaki, and T. Harada, "IBC127: Video dataset for fine-grained bird classification," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Seattle, WA, USA, Jul. 2016, pp. 1–6, doi: 10.1109/ICME.2016.7552915.

[14] C. Pang, H. Li, A. Cherian, and H. Yao, "Part-based fine-grained bird image retrieval respecting species correlation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Beijing, China, Sep. 2017, pp. 2896–2900, doi: 10.1109/ICIP.2017.8296812.

[15] J. Liu, A. Kanazawa, D. Jacobs, and P. Belhumeur, "Dog breed classification using part localization," in *Proc. Eur. Conf. Comput. Vis.* vol. 7572. Berlin, Germany: Springer, 2012, pp. 172–185, doi: 10.1007/978-3-642-33718-5_13.

[16] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, and Z. Zhang, "The application of two-level attention models in deep convolutional neural network for fine-grained image classification," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 842–850, doi: 10.1109/CVPR.2015.7298685.

[17] J. Fu, H. Zheng, and T. Mei, "Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 4476–4484, doi: 10.1109/CVPR.2017.476.

[18] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Proc. 6th Indian Conf. Comput. Vis., Graph. Image Process.*, Bhubaneswar, India, Dec. 2008, pp. 722–729, doi: 10.1109/ICVGIP.2008.47.

[19] H. Zheng, J. Fu, T. Mei, and J. Luo, "Learning multi-attention convolutional neural network for fine-grained image recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 5219–5227, doi: 10.1109/ICCV.2017.557.

[20] M. Chevalier, N. Thome, M. Cord, J. Fournier, G. Henaff, and E. Dusch, "LR-CNN for fine-grained classification with varying resolution," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Quebec City, QC, Canada, Sep. 2015, pp. 3101–3105, doi: 10.1109/ICIP.2015.7351374.

[21] C. Zhang, C. Liang, L. Li, J. Liu, Q. Huang, and Q. Tian, "Fine-grained image classification via low-rank sparse coding with general and class-specific codebooks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 7, pp. 1550–1559, Jul. 2017, doi: 10.1109/TNNLS.2016.2545112.

[22] P. Zhang, F. Wang, and Y. Zheng, "Self supervised deep representation learning for fine-grained body part recognition," in *Proc. IEEE 14th Int. Symp. Biomed. Imag. (ISBI)*, Melbourne, VIC, Australia, Apr. 2017, pp. 578–582, doi: 10.1109/ISBI.2017.7950587.

[23] J. Liang, J. Guo, X. Liu, and S. Lao, "Fine-grained image classification with Gaussian mixture layer," *IEEE Access*, vol. 6, pp. 53356–53367, 2018, doi: 10.1109/ACCESS.2018.2871621.

[24] A. Iscen, G. Tolias, P.-H. Gosselin, and H. Jégou, "A comparison of dense region detectors for image search and fine-grained classification," *IEEE Trans. Image Process.*, vol. 24, no. 8, pp. 2369–2381, Aug. 2015, doi: 10.1109/TIP.2015.2423557.

[25] Q. Xuan, H. Xiao, C. Fu, and Y. Liu, "Evolving convolutional neural network and its application in fine-grained visual categorization," *IEEE Access*, vol. 6, pp. 31110–31116, 2018, doi: 10.1109/ACCESS.2018.2842202.

[26] Y. Seo and K.-S. Shin, "Image classification of fine-grained fashion image based on style using pre-trained convolutional neural network," in *Proc. IEEE 3rd Int. Conf. Big Data Anal. (ICBDA)*, Shanghai, China, Mar. 2018, pp. 387–390, doi: 10.1109/ICBDA.2018.8367713.

[27] Q. Zhang, L. Zhuo, S. Zhang, J. Li, H. Zhang, and X. Li, "Fine-grained vehicle recognition using lightweight convolutional neural network with combined learning strategy," in *Proc. IEEE 4th Int. Conf. Multimedia Big Data (BigMM)*, Xi'an, China, Sep. 2018, pp. 1–5, doi: 10.1109/BigMM.2018.8499085.

[28] L. Zhang, Y. Gao, Y. Xia, Q. Dai, and X. Li, "A fine-grained image categorization system by cellet-encoded spatial pyramid modeling," *IEEE Trans. Ind. Electron.*, vol. 62, no. 1, pp. 564–571, Jan. 2015, doi: 10.1109/TIE.2014.2327558.

[29] C. Wah, S. Branson, P. Perona, and S. Belongie, "Multiclass recognition and part localization with humans in the loop," in *Proc. Int. Conf. Comput. Vis.*, Barcelona, Spain, Nov. 2011, pp. 2524–2531, doi: 10.1109/ICCV.2011.6126539.

[30] J. Deng, J. Krause, and L. Fei-Fei, "Fine-grained crowdsourcing for fine-grained recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Portland, OR, USA, Jun. 2013, pp. 580–587, doi: 10.1109/CVPR.2013.81.

[31] Y. Zhang *et al.*, "Weakly supervised fine-grained categorization with part-based image representation," *IEEE Trans. Image Process.*, vol. 25, no. 4, pp. 1713–1725, Apr. 2016, doi: 10.1109/TIP.2016.2531289.

[32] L. Li, Y. Guo, L. Xie, X. Kong, and Q. Tian, "Fine-grained visual categorization with fine-tuned segmentation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Quebec City, QC, Canada, Sep. 2015, pp. 2025–2029, doi: 10.1109/ICIP.2015.7351156.

[33] E. Gavves, B. Fernando, C. G. M. Snoek, A. W. M. Smeulders, and T. Tuytelaars, "Fine-grained categorization by alignments," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sydney, NSW, Australia, Dec. 2013, pp. 1713–1720, doi: 10.1109/ICCV.2013.215.

[34] G. Chen, J. Yang, H. Jin, E. Shechtman, J. Brandt, and T. X. Han, "Selective pooling vector for fine-grained recognition," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Waikoloa, HI, USA, Jun. 2015, pp. 860–867, doi: 10.1109/WACV.2015.119.

[35] A. Angelova and S. Zhu, "Efficient object detection and segmentation for fine-grained recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Portland, OR, USA, Jun. 2013, pp. 811–818, doi: 10.1109/CVPR.2013.110.

[36] A. Angelova and P. M. Long, "Benchmarking large-scale fine-grained categorization," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Steamboat Springs, CO, USA, Mar. 2014, pp. 532–539, doi: 10.1109/WACV.2014.6836056.

[37] G. Sun, Y. Chen, X. Liu, and E. Wu, "Adaptive multi-task learning for fine-grained categorization," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Quebec City, QC, Canada, Sep. 2015, pp. 996–1000, doi: 10.1109/ICIP.2015.7350949.

[38] G. Zheng, M. Tan, J. Yu, Q. Wu, and J. Fan, ''Fine-grained image recognition via weakly supervised click data guided bilinear CNN model,'' in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Hong Kong, Jul. 2017, pp. 661–666, doi: 10.1109/ICME.2017.8019407.

[39] L. Zhang, Y. Cao, X. Xiang, and N. U. R. Junejo, ''Efficient match kernel in fine-grained image categorization,'' in *Proc. 11th World Congr. Intell. Control Autom.*, Shenyang, China, Jun./Jul. 2014, pp. 5578–5581, doi: 10.1109/WCICA.2014.7053669.

[40] M. Srinivas, Y.-Y. Lin, and H.-Y. M. Liao, ''Deep dictionary learning for fine-grained image classification,'' in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Beijing, China, Sep. 2017, pp. 835–839, doi: 10.1109/ICIP.2017.8296398.

[41] B. Yao, G. Bradski, and L. Fei-Fei, ''A codebook-free and annotation-free approach for fine-grained image categorization,'' in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Providence, RI, USA, Jun. 2012, pp. 3466–3473, doi: 10.1109/CVPR.2012.6248088.

[42] H. Yao, S. Zhang, Y. Zhang, J. Li, and Q. Tian, ''Coarse-to-fine description for fine-grained visual categorization,'' *IEEE Trans. Image Process.*, vol. 25, no. 10, pp. 4858–4872, Oct. 2016, doi: 10.1109/TIP.2016.2599102.

[43] H. Yao, S. Zhang, C. Yan, Y. Zhang, J. Li, and Q. Tian, ''AutoBD: Automated Bi-level description for scalable fine-grained visual categorization,'' *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 10–23, Jan. 2018, doi: 10.1109/TIP.2017.2751960.

[44] X. Bai, M. Yang, P. Lyu, Y. Xu, and J. Luo, ''Integrating scene text and visual appearance for fine-grained image classification,'' *IEEE Access*, vol. 6, pp. 66322–66335, 2018, doi: 10.1109/ACCESS.2018.2878899.

[45] S. Karaoglu, R. Tao, J. C. van Gemert, and T. Gevers, ''Context: Text detection for fine-grained object classification,'' *IEEE Trans. Image Process.*, vol. 26, no. 8, pp. 3965–3980, Aug. 2017, doi: 10.1109/TIP.2017.2707805.

[46] L. Liao, R. Hu, J. Xiao, Q. Wang, J. Xiao, and J. Chen, ''Exploiting effects of parts in fine-grained categorization of vehicles,'' in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Quebec City, QC, Canada, Sep. 2015, pp. 745–749, doi: 10.1109/ICIP.2015.7350898.

[47] S. Xie, T. Yang, X. Wang, and Y. Lin, ''Hyper-class augmented and regularized deep learning for fine-grained image classification,'' in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 2645–2654, doi: 10.1109/CVPR.2015.7298880.

[48] X. Wang, R. Li, and J. Currey, ''Leveraging 2D and 3D cues for fine-grained object classification,'' in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Phoenix, AZ, USA, Sep. 2016, pp. 1354–1358, doi: 10.1109/ICIP.2016.7532579.

[49] L. Zhang, Y. Yang, M. Wang, R. Hong, L. Nie, and X. Li, ''Detecting densely distributed graph patterns for fine-grained image categorization,'' *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 553–565, Feb. 2016, doi: 10.1109/TIP.2015.2502147.

[50] Y. Hu, K. Li, and H. Zhang, ''Cross-modal face matching: Tackling visual abstraction using fine-grained attributes,'' in *Proc. Vis. Commun. Image Process. (VCIP)*, Chengdu, China, Nov. 2016, pp. 1–4, doi: 10.1109/VCIP.2016.7805451.

[51] W. Di, C. Wah, A. Bhardwaj, R. Piramuthu, and N. Sundaresan, ''Style finder: Fine-grained clothing style detection and retrieval,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Portland, OR, USA, Jun. 2013, pp. 8–13, doi: 10.1109/CVPRW.2013.6.

[52] X. Ou, Z. Wei, H. Ling, S. Liu, and X. Cao, ''Deep multi-context network for fine-grained visual recognition,'' in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Seattle, WA, USA, Jul. 2016, pp. 1–4, doi: 10.1109/ICMEW.2016.7574666.

[53] H. Lu, Z. Cao, Y. Xiao, Z. Fang, and Y. Zhu, ''Toward good practices for fine-grained maize cultivar identification with filter-specific convolutional activations,'' *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 2, pp. 430–442, Apr. 2018, doi: 10.1109/TASE.2016.2616485.

[54] H. Lu, Z. Cao, Y. Xiao, Z. Fang, and Y. Zhu, ''Fine-grained maize cultivar identification using filter-specific convolutional activations,'' in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Phoenix, AZ, USA, Sep. 2016, pp. 3718–3722, doi: 10.1109/ICIP.2016.7533054.

[55] Z. Lin, S. Mu, A. Shi, C. Pang, and X. Sun, ''A novel method of maize leaf disease image identification based on a multichannel convolutional neural network,'' *Trans. ASABE*, vol. 61, no. 5, pp. 1461–1474, 2018, doi: 10.13031/trans.12440.

[56] J. Zhang, J. Luo, W. Huang, and J. Wang, ''Continuous wavelet analysis based spectral features selection for winter wheat yellow rust detection,'' in *Proc. World Autom. Congr. (WAC)*, Puerto Vallarta, Mexico, Jun. 2012, pp. 195–200, doi: 10.1080/10798587.2011.10643167.

[57] M. Wafy, H. Ibrahim, and E. Kamel, ''Identification of weed seeds species in mixed sample with wheat grains using SIFT algorithm,'' in *Proc. 9th Int. Comput. Eng. Conf. (ICENCO)*, Giza, Egypt, Dec. 2013, pp. 11–14, doi: 10.1109/ICENCO.2013.6736468.

[58] J. Li, L. Gao, and Z. Shen, ''Extraction and analysis of digital images feature of three kinds of wheat diseases,'' in *Proc. 3rd Int. Congr. Image Signal Process.*, Yantai, China, Oct. 2010, pp. 2543–2548, doi: 10.1109/CISP.2010.5646912.

[59] R. V. Ronge and M. M. Sardeshmukh, ''Comparative analysis of Indian wheat seed classification,'' in *Proc. 9th Int. Comput. Eng. Conf. (ICENCO)*, New Delhi, India, Sep. 2014, pp. 937–942, doi: 10.1109/ICACCI.2014.6968483.

[60] E. O. Güneş, S. Aygün, M. Kırcı, A. Kalateh, and Y. Çakır, ''Determination of the varieties and characteristics of wheat seeds grown in Turkey using image processing techniques,'' in *Proc. 3rd Int. Conf. Agro-Geoinformatics*, Beijing, China, Aug. 2014, pp. 1–4, doi: 10.1109/Agro-Geoinformatics.2014.6910610.

[61] L. Wang, F. Dong, Q. Guo, C. Nie, and S. Sun, ''Improved rotation kernel transformation directional feature for recognition of wheat stripe rust and powdery mildew,'' in *Proc. 7th Int. Congr. Image Signal Process.*, Dalian, China, Oct. 2014, pp. 286–291, doi: 10.1109/CISP.2014.7003793.

[62] J. Zhao, J. Guo, C. Liu, D. Zhang, and L. Huang, ''Monitoring of powdery mildew on Winter wheat using multi-temporal HJ-CCD imagery on a regional scale,'' in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Fort Worth, TX, USA, Jul. 2017, pp. 5085–5088, doi: 10.1109/IGARSS.2017.8128146.

[63] Y. Tian, C. Zhao, S. Lu, and X. Guo, ''SVM-based multiple classifier system for recognition of wheat leaf diseases,'' in *Proc. World Autom. Congr. (WAC)*, Puerto Vallarta, Mexico, Jun. 2012, pp. 189–193, doi: 10.1080/10798587.2011.10643166.

[64] X. Niu, M. Wang, X. Chen, S. Guo, H. Zhang, and D. He, ''Image segmentation algorithm for disease detection of wheat leaves,'' in *Proc. Int. Conf. Adv. Mechatronics Syst.*, Kumamoto, Japan, Aug. 2014, pp. 270–273, doi: 10.1109/ICAMechS.2014.6911663.

[65] X. Yang, H. Yang, W. Huang, C. Li, X. Xu, and J. Wang, ''Study of spatial-temporal spread model for wheat stripe rust in small scale based on Bayesian network,'' in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Munich, Germany, Jul. 2012, pp. 503–506, doi: 10.1109/IGARSS.2012.6351374.

[66] D. H. Hubel and T. N. Wiesel, ''Receptive fields, binocular interaction and functional architecture in the cat's visual cortex,'' *J. Physiol.*, vol. 160, no. 1, pp. 106–154, 1962, doi: 10.1113/jphysiol.1962.sp006837.

[67] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, ''Gradient-based learning applied to document recognition,'' *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998, doi: 10.1109/5.726791.

[68] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ''ImageNet classification with deep convolutional neural networks,'' in *Proc. Adv. Neural Inf. Process. Syst.*, Lake Tahoe, NV, USA: Curran Associates, Inc., vol. 60, 2012, pp. 1097–1105, doi: 10.1145/3065386.

[69] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, ''ImageNet: A large-scale hierarchical image database,'' in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Miami, FL, USA, Jun. 2009, pp. 248–255, doi: 10.1109/CVPR.2009.5206848.

[70] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, ''DeepFace: Closing the gap to human-level performance in face verification,'' in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Columbus, OH, USA, Jun. 2014, pp. 1701–1708, doi: 10.1109/CVPR.2014.220.

[71] W. Ouyang *et al.*, ''DeepID-Net: Deformable deep convolutional neural networks for object detection,'' in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 2403–2412, doi: 10.1109/CVPR.2015.7298854.

[72] C. Szegedy *et al.*, ''Going deeper with convolutions,'' in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.

[73] K. Simonyan and A. Zisserman, ''Very deep convolutional networks for large-scale image recognition,'' in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Apr. 2015, pp. 1–14.

[74] K. He, X. Zhang, S. Ren, and J. Sun, ''Deep residual learning for image recognition,'' in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[75] D. Silver *et al.*, ''Mastering the game of Go with deep neural networks and tree search,'' *Nature*, vol. 529, no. 7587, pp. 484–489, 2016, doi: 10.1038/nature16961.

[76] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," in *Proc. Int. Conf. Represent. Learn. (ICLR)*, Nov. 2014, pp. 1–16.

[77] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.* vol. 8689. Cham, Switzerland: Springer, Nov. 2014, pp. 818–833, doi: 10.1007/978-3-319-10590-1_53.

[78] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 580–587, doi: 10.1109/CVPR.2014.81.

[79] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[80] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?" in *Proc. IEEE 12th Int. Conf. Comput. Vis. (ICCV)*, Kyoto, Japan, Sep./Oct. 2009, pp. 2146–2153, doi: 10.1109/ICCV.2009.5459469.

[81] B. Chen, W. Deng, and J. Du, "Noisy softmax: Improving the generalization ability of DCNN via postponing the early softmax saturation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4021–4030, doi: 10.1109/CVPR.2017.428.

[82] X. Li *et al.*, "Supervised latent Dirichlet allocation with a mixture of sparse softmax," *Neurocomputing*, vol. 312, pp. 324–335, Oct. 2018, doi: 10.1016/j.neucom.2018.05.077.

[83] N. Dawar, S. Ostadabbas, and N. Kehtarnavaz, "Data augmentation in deep learning-based fusion of depth and inertial sensing for action recognition," *IEEE Sensors Lett.*, Oct. 2018, doi: 10.1109/LSENS.2018.2878572.

[84] A. Fawzi, H. Samulowitz, D. Turaga, and P. Frossard, "Adaptive data augmentation for image classification," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Phoenix, AZ, USA, Sep. 2016, pp. 3688–3692, doi: 10.1109/ICIP.2016.7533048.

[85] R. Dellana and K. Roy, "Data augmentation in CNN-based periocular authentication," in *Proc. 6th Int. Conf. Inf. Commun. Manage. (ICICM)*, Hatfield, U.K., Oct. 2016, pp. 141–145, doi: 10.1109/INFOCOMAN.2016.7784231.

[86] P.-T. de Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Ann. Oper. Res.*, vol. 134, no. 1, pp. 19–67, 2005, doi: 10.1007/s10479-005-5724-z.

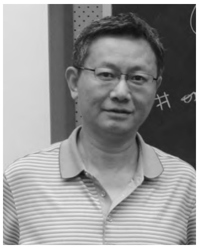
[87] C. H. Li and P. K. S. Tam, "An iterative algorithm for minimum cross entropy thresholding," *Pattern Recognit. Lett.*, vol. 19, no. 8, pp. 771–776, 1998, doi: 10.1016/S0167-8655(98)00057-9.

[88] X. Liu, Z. Shi, X. Zhang, and C. Yang, "A batch normalization autoencoder model for breast cancer multidimensional follow-up data," in *Proc. IEEE Int. Conf. Smart Internet Things (SmartIoT)*, Xi'an, China, Aug. 2018, pp. 178–185, doi: 10.1109/SmartIoT.2018.00040.

[89] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1520–1528, doi: 10.1109/ICCV.2015.178.

[90] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 447–456, doi: 10.1109/CVPR.2015.7298642.

ZHONGQI LIN is currently pursuing the Ph.D. degree with the College of Information and Electrical Engineering, China Agricultural University, Beijing, China. His current research interests include deep learning for image classification, deep transfer learning for object recognition and segmentation, deep reinforcement learning for prediction, natural language processing, machine learning, data-intensive parallel programming, and digital image processing techniques, including enhancement, compression, and denoising, with specific interests in object classification.



SHAOMIN MU received the M.S. degree in computer application technology from the Shandong University of Science and Technology, Taian, China, in 2000, and the Ph.D. degree in computer application technology from Beijing Jiaotong University, Beijing, China, in 2008. He is currently a Professor with the College of Information Science and Engineering, Shandong Agricultural University. He has published 30 academic papers in domestic and foreign journals, and among them, over 15 are cited by SCI/EI/ISTP. He has also participated in sub-projects of the National 973 Program and the National Natural Funds. His current research interests include machine learning, pattern recognition, digital image processing techniques, computer vision, data-intensive parallel programming, and their applications in agriculture and other fields.

FENG HUANG received the Ph.D. degree from the Institute of Physics, Chinese Academy of Sciences, in 2005. She is currently a Professor with the College of Science, China Agricultural University.

KHATTAK ABDUL MATEEN received the Ph.D. degree in horticulture and landscape from the University of Reading, U.K., in 1999. He was a Research Scientist in different agriculture research organizations before joining Agricultural University Peshawar, Pakistan, where he is currently a Professor with considerable experience in teaching and research at the Department of Horticulture. He has conducted academic and applied research on different aspects of tropical fruits, vegetables, and ornamental plants. He was also with Alberta Agriculture and Forestry, Canada, as a Research Associate, and with the Organic Agriculture Centre of Canada as a Research and Extension Coordinator (for Alberta province). There he helped in developing organic standards for greenhouse production and energy saving technologies for Alberta greenhouses. He is also a Visiting Professor with the College of Information and Electrical Engineering, China Agricultural University, Beijing. He has published 55 research articles in scientific journals of international repute. His research interests include greenhouse production, medicinal, aromatic and ornamental plants, light quality, supplemental lighting and temperature effects on greenhouse crops, aquaponics, and organic production.

MINJUAN WANG received the Ph.D. degree from the School of Biological Science and Medical Engineering, Beihang University, under the supervision of Prof. Hong Liu, in 2017. She was a Visiting Scholar with the School of Environmental Science, Ontario Agriculture College, University of Guelph, from 2015 to 2017. She is currently a Postdoctoral Fellow from the School of Information and Electrical Engineering, China Agricultural University. Her research interests include bioinformatics and the Internet of Things key technologies.

WANLIN GAO received the B.S., S.M., and Ph.D. degrees from China Agricultural University, in 1990, 2000, and 2010, respectively. He is currently the Dean of the College of Information and Electrical Engineering, China Agricultural University. His research interests include the informationization of new rural areas, intelligence agriculture, and the service for rural comprehensive information. He is also a member of the Science and Technology Committee of the Ministry of Agriculture, a member of the Agriculture and Forestry Committee of Computer Basic Education in Colleges and Universities, a Senior Member of the Society of Chinese Agricultural Engineering, and so on.

He has been a Principal Investigator of over 20 national plans and projects. He has published 90 academic papers in domestic and foreign journals, and among them, over 40 are cited by SCI/EI/ISTP. He has written two teaching materials, which are supported by the National Key Technology R&D Program of China during the 11th Five-Year Plan Period, and has written five monographs. Moreover, he holds 101 software copyrights, 11 patents for inventions, and eight patents for new practical inventions.

JINGDUN JIA is currently a Researcher with the China Rural Technology Development Center. He is also a Committee Member of the Policy Advisory Board for the Australian Center for International Agricultural Research and an Adjunct Professor with China Agricultural University. He has long been involved in development strategy, plan, and policy for science and technology management. His research interests include agricultural and rural development, and regional development strategy. He has conducted in-depth research on rural scientific and technological innovation, agricultural biotechnology and food industry, biological energy and biomass industry, nutrition and health, and intelligent agricultural scientific and technological innovation.