Multi-stream aggregation network for fine-grained crop pests and diseases image recognition

Xuebo Jin, Zhi Tao and Jianlei Kong*

Beijing Technology and Business University,
Beijing, China
Email: jinxuebo@btbu.edu.cn
Email: taozhi@st.btbu.edu.cn
Email: kongjianlei@btbu.edu.cn
*Corresponding author

Abstract: Pests and disease recognition can be considered as Fine-Grained Visual Classification (FGVC) problems, suffering low inter-class discrepancy and high intra-class variances from the sub-categories, which is more challenging than common basic-level category classification dependent on traditional Deep Neural Networks (DNNs). To encourage further progress in challenging realistic agricultural conditions, this paper presents a realistic CropDP181 data set with 181 categories and fine-grained multi-stream aggregation network with models transferred named as MSA-NET (Multi-Stream Aggregation Network) for fine-grained species recognition based on fusion idea. The novel MSA-NET model combines ResNet, NTS-Net (Navigator-Teacher-Scrutiniser Network), and FAST-MPN-COV (Towards Faster Training of Global Covariance Pooling Network) trained by a multi-stream feature extractor to exploit the high-dimensional feature maps representing discriminative and non-discriminative parts as well as interclass variances. Finally, a fusion module equipped with NetVLAD (Network Vector of Locally Aggregated Descriptors) layer is developed to fuse different components model as a unified probability representation for the ultimate fine-grained recognition. The MSA-NET model achieves competitive results in fine-grained pests and disease recognition outperforming state-of-the-art methods.

Keywords: crop pests and diseases; fine-grained visual classification; multi-stream neural networks; NetVLAD aggregation.

Reference to this paper should be made as follows: Jin, X., Tao, Z. and Kong, J. (2021) 'Multi-stream aggregation network for fine-grained crop pests and diseases image recognition', *Int. J. Cybernetics and Cyber-Physical Systems*, Vol. 1, No. 1, pp.52–67.

Biographical notes: Xuebo Jin is a Professor of Control Science and Engineering at Beijing Technology and Business University, Supervisor of Master's degree, Member of Information Fusion Branch of CAAC, Member of Intelligent Products and Industry Working Committee of CAAI. Her research interests include information fusion, big data analysis, state estimation, video tracking, etc.

Zhi Tao received his Master's degree from Beijing Technology and Business University. His research direction is computer vision.

Jianlei Kong is an Associate Professor from Beijing Technology and Business University. His research interests include fusion, statistical signal processing of mutli-sensors including computer vision, 3D reconstruction and classification from large-scale point clouds, pattern recognition and artificial intelligence.

1 Introduction

For agricultural countries, especially those developing countries whose economic growth depends mainly on agriculture, the health status of crops is a crucial point in agricultural research; in the growth of crops, pests and diseases are the natural enemies of most agricultural plants; therefore, research on pests and diseases is a key link to protect crop growth. In reality, a crop may experience a range of pests and diseases. However, pests and diseases may occur in many different parts, such as roots, shoots, stems, leaves, and fruits of plants; the fact that crops are attacked by a wide variety of pests and diseases makes the applicability of computer vision technology in the correct identification of pests and diseases more difficult, especially for the rapid development of intelligent agriculture today.

With the rapid development of the Internet of Things (IoT) and deep learning, the monitoring and shooting of crops have become more and more convenient. The information acquisition of diseases and pests has become more accurate, which contributes to the diagnosis and identification of pests and diseases. In fact, rapid development of deep learning makes image recognition easier to achieve, but in complex agricultural contexts, especially on pests and diseases, the actual situation of pests and diseases in agriculture is complex; such as multiple periods of goals, inter-class similarity and intra-class differences. Therefore, existing DNNs-based methods achieving state-of-the-art performance on other research fields, such as VGGNet (Simonyan and Zisserman, 2014), ResNet (He et al., 2016) and DenseNet (Huang et al., 2017), are not suitable for Fine-Grained Visual Classification (FGVC) tasks of pests and diseases. Considering these situations, the popular fine-grained image classification model in the field of image recognition provides us with a better choice. Actually, FGVC remains a challenging task and more difficult than common image classification because objects from similar subordinate categories may have marginal visual differences that are difficult to distinguish by traditional DNNs or even humans. In reality, the identification of diseases and insect pests has these three fine-grained complex situations: plant disease or pest has multiple periods, inter-class similarity, and intra-class difference. It limits the further application of deep-learning technology in various agricultural missions and the wider development of computer vision.

To distinguish fine-grained categories with a very similar outline, it requires specialised knowledge focusing on feature representation of discriminative object parts to expand the application of existing DNNs on FGVC. According to whether the method requires additional part location annotation, current state-of-the-arts can be divided into two groups: Strongly-Supervised Learning (SSL) and Weakly-Supervised Learning (WSL) (Han et al., 2015). The so-called SSL refers to the use of additional manual annotation information, such as object annotation frame and position annotation points, to obtain better classification accuracy in model training. For example, Zhang et al.

(2014) proposed a part-based *R*-CNN fine-grained classification model, which uses *R*-CNN algorithm to detect object level (e.g. birds) and its local areas (head, body, etc.) in fine-grained images. However, part-based *R*-CNN needs the help of bounding box and part annotation in training, and to achieve satisfactory classification accuracy, it also requires the testing image to provide bounding box, which limits the application of part-based R-CNN in actual scenes. Sensed by part-based *R*-CNN, Branson et al. (2014) proposed that the detection boxes of object level and part level could be obtained after the prediction points of part annotation were obtained by DPM algorithm. Thus, another fine-grained classification algorithm Pose Normalised CNN was developed. Nevertheless, it is also dependent on extra location annotation with expensive manual labelling, which makes it hard to be prevalently applied in practice.

Weak supervised fine-grained image classification is the mainstream method of fine-grained classification at present because no additional labelling cost is required. For example, Xiao et al. (2015) proposed the Two-Level Attention models (TLAN) to extract object-level and part-level features in bottom-to-up way at the same time. Then spectral clustering is employed to select important semantic parts of two-level attention for finding the discriminative area between the foreground object and parts. Lin et al. (2015) designed a novel bilinear network model (Bilinear CNN) is to combine two stream features at each location using the outer product, which considers their pairwise interactions in the end-to-end training process. Similarly, Peng et al. (2017) proposed the Object-Part Attention Model (OPAM) for weakly supervised fine-grained image classification without either object or part annotations, which avoids the heavy labour consumption of labelling. This model integrates two level attention: object-level attention localises objects of images, and part-level attention selects discriminative parts. Both are jointly employed to learn multi-view and multi-scale features to enhance their mutual promotion. Spatial Transformer Network (ST-CNN) (Jaderberg et al., 2015) also chooses a weakly-supervised way. The model can also locate several object parts simultaneously to achieve more accurate classification performance by first learning a proper geometric transformation and align the image before classifying.

In the training process, the former SSL requires additional location information apart from image-level category labels, such as bounding-boxes or key-points of discriminative parts. Location annotation heavily relies on more expensive manual labelling and time-cost, which makes it hard to be prevalently applied in practice. As a consequence, researchers pay more attention to WSL frameworks, which only employ image-level annotation to achieve FGVC tasks. For instance, the attention mechanism can be implemented to capture local features in a translationally invariant manner, which is particularly suitable for classifying fine-grained categories without manual location annotations. However, these WSL methods universally suffer lower performance than the best SSL models, especially when small objects appear in a cluttered background.

Moreover, given the learned location features of objects' parts, a single WSL model is likely to focus on the constant architecture of parts distribution and cannot distinguish interclass variances between similar fine-grained classes. More importantly, diverse WSL models are interested in multiple object parts with different preferences, which aggravate intra-class deviations of the same class. Consequently, it is very likely to bring about the wrong category when these parts are occluded due to pose or viewpoint variances, which leads to the diverse recognition performance of different WSL models.

In this article, to integrate multiple models' advantages for discovering discriminative parts, an active multi-stream aggregation network named as MSA-NET is designed to utilise the mixture-granularity information of multiple DNNs by only using image-level labels. First, the MSA-NET generates input images with various data that augment pre-processing on the data set we created, in which all images are collected by different cameras and equipment of IoT outdoors. The framework is then trained by the multi-stream DNNs architecture to exploit the high-dimensional feature maps representing discriminative and non-discriminative parts as well as interclass variances. Finally, a NetVLAD-aggregation module is developed to fuse different features as a unified representation for the ultimate fine-grained recognition. This optimisation design offers a high capacity to learn complementary yet correlated information for intra-class variances among multi-grained feature maps of different models, making the proposed MSA-NET more suitable for fine-grained pests and diseases species recognition. The experimental results show the robustness and superiority of MSA-NET.

2 Data set overview

In research, agricultural machines and robots independently collect images of crops and immediately convert them into management measures, allow for a high level of spatial and seasonal dynamics. Those field maintenance tasks severely dependent on the real-time performance of online decision-making algorithms and stored in the platform. Subsequently, the multiple devices collected the crops images to meet more complex tasks of farmers and agronomists, which allow not only to monitor the health and growth of the crops continuously, but also to determine the operation measures for autonomous robots. Moreover, the images of smartphones play a significant role in social contact and sharing. Therefore, it is very challenging to classify and detect crop pests and disease species from large images with different angles, focal lengths, and resolutions, offered by various devices in the platform.

Based on the above data acquisition platform, the process platform of its complete deep learning model is shown in Figure 1. With the concept of wisdom agriculture, the combination of agricultural internet technology and deep learning is an effective way to build computer vision problems in agriculture quickly. Based on various devices and equipment, the data set in this paper collects 124,437 images including crop diseases and pests of 88 upper-level categories and 181 sub-classes, which are the most reasonable for the PA purpose. Among them, the diseases were collected from 11 crops such as alfalfa, corn and tomato. The pests originated from 77 families, such as butterflies and bees. According to statistics, there are 100 in the least number of categories and 5109 in the largest number. The size of the data set is sufficient to meet the training requirement. Moreover, the condition that the data set has 181 classes has a sound premise basis for fine-grained classification.

In the data set, multiple patterns of the same disease or insect pest, similar forms of different categories are shown in Figure 2. As can be seen from the figure, they all have many similarities, and there are countless similar scenarios in a wide variety of pests and diseases. When extracting the features of these images, the common deep neural networks usually only extract their common features; after training, some similar but not the same kind of test images with different shapes will be misclassified or confused, which leads to the difficulty of improving the accuracy of classification and recognition.

These two phenomena are the roadblocks that hinder the classification of pests and diseases to higher precision, but they are the best experimental materials for fine-grained classification models.

Figure 1 Complete data and learning platform

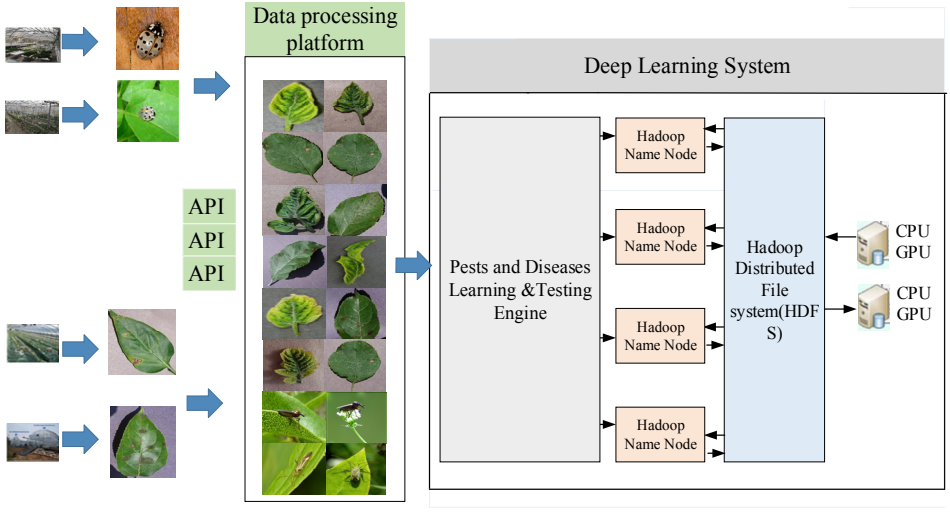


Figure 2 Fine-grained image samples of crop diseases and pests (a) Crop disease sample (b) Crop pest samples

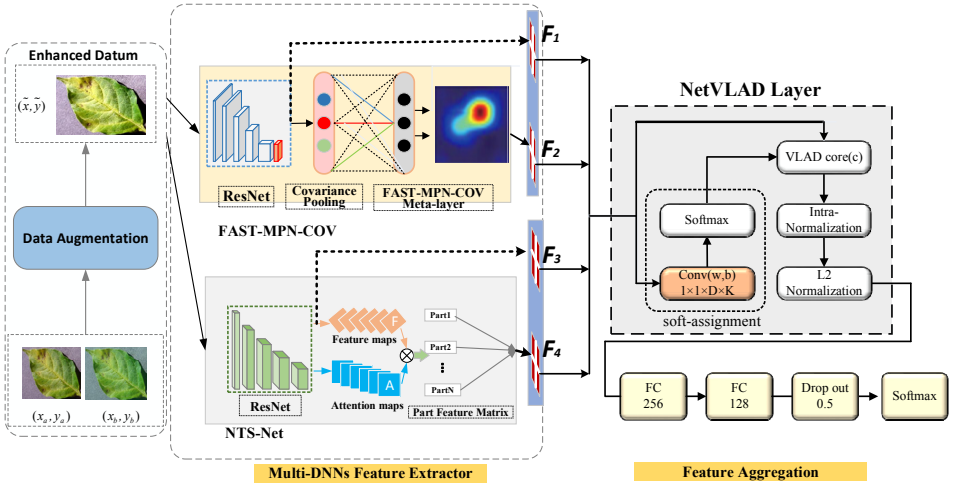
(a)

(b)

3 Multi-stream aggregation network architecture

By analysing the differences in the recognition ability of individual models for different categories and the recognition of multiple models in the same category, we find that there are differences between different models and different feature extraction capabilities for pictures. Therefore, we design a fusion strategy to fuse the confidence results of the model through the fully connected layer output, so that the model MSA-NET complement each other and improve the accuracy, as shown in Figure 3.

Figure 3 Schematic diagram of the MSA-NET model



3.1 Data augmentation

To avoid over-fitting of the network, some data augmentations are applied to enhance a larger number of data set images with high quality before training. Generally, we need to resize the input images with various adjusted forms and distortion before sending them to the network. Hence, the more complicated the tasks are, the more images the DNN models need to nonlinearly estimate massive parameters adopted by most classification works, especially when using images with low resolution. To address this problem, we formulate a series of augmented methods to increase the general training datum, which consists of the following four steps one-by-one:

1) Randomly crop a rectangular region whose aspect ratio is randomly sampled in [3/4; 4/3] and area randomly sampled in [8%, 100%], then resize the cropped region into a 448-by-448 square image.

2) Randomly flip each image 180° horizontally and vertically with a probability of 0.5 probability to increase the image's diversity. Randomly rotate each image in 90, 180, and 270 clockwise to improve distortion adaptability.

3) In the HSV colour space of the image, exponentially changing the saturation S and brightness V components of each pixel, keeping the hue H constant, to increasing the illumination variation. The S and V channels are respectively scaled with coefficients

uniformly drawn from [0.25, 4]. Randomly sample an image and decode it into 32-bit floating point raw pixel values in [0, 255]. And add PCA noise with a coefficient sampled from a normal distribution $N(0, 0.1)$.

4) Finally, the mixup augmentation method proposed in Zhang et al. (2017) is selected to regularise the network models to favour simple linear behaviour in-between training examples for alleviating undesirable behaviours. In the mixup step, each time two examples are randomly sampled from training data to form a new virtual training example by a weighted linear interpolation:

$$\begin{aligned}\tilde{x}&=\lambda x_a+(1-\lambda)x_b\\ \tilde{y}&=\lambda y_a+(1-\lambda)y_b\end{aligned} \tag{1}$$

where x_a and x_b are raw input image vectors, y_a and y_b are one-hot encoding labels, λ is a random number drawn from the beta (α,α) distribution with the value range [0,1].

The mixup hyper-parameter α controls the strength of interpolation between vector-label pairs, of which value is recommended as tending to 0. In this paper, we set $\alpha = 0.18$ in the beta distribution and increase the epoch number asking for longer training progress to converge better performance. Thus, we achieve additional high-quality examples $(\tilde{x},\tilde{y})$ through the enhanced data augmentation for subsequent model training. Those above steps can obtain improved generalisation and robustness abilities of the network architecture by the augmented datum.

3.2 Multi-steam feature extractor

The feature extractor of the proposed MSA-NET consists of multiple classification deep neural networks that are trained concurrently on the augmented datum from the first stage. In this section, we will use NTS-Net (Yang et al., 2018) and FAST-MPN-COV (Li et al., 2018) models to obtain multi-dimensional feature maps. For each sub-model, we make some minor adjustments to the network architecture, such as changing the stride or kernel size of a particular convolution layer. Such a tweak often barely changes the computational complexity but might have a non-negligible effect on the model accuracy.

Firstly, the FAST-MPN-COV is employed to extract fine-grained feature maps of small-scale object's parts. This model is an iterative matrix square root normalisation method for fast end-to-end training of global covariance pooling networks, which consists of a basic classification network (AlexNet or ResNet), some covariance pooling layers and an FAST-MPN-COV meta-layer. To further improve our proposed architecture's performance and efficiency, we adopt the transfer-learning strategy to learn the professional representation capability of the object's parts based on the coarse-grained domain knowledge from the ResNet model as mentioned above. We transfer the trained ResNet50 network as the classification network of FAST-MPN-COV, which avoids the repeated parameter calculation in much fewer epochs, further accelerating network training. After the last convolutional layer of ResNet50, we add some 1 × 1 convolution with c_1=256 channels to down sample the outputted feature tensor, which outputs a 14 × 14 × 256 tensor. Then a second-order pooling is performed to estimate the

covariance matrix. Subsequently, the model designs a meta-layer with loop-embedded directed graph structure for computing approximate square root of the covariance matrix. The meta-layer consists of three nonlinear structured layers, performing pre-normalisation, coupled Newton-Schulz iteration and post-compensation, respectively. The first pre-normalisation layer guarantees the following iteration convergence, which is achieved by dividing the covariance matrix by its trace. The second layer is of the loop structure, repeating the coupled matrix equations involved in Newton-Schulz iteration a fixed number of times, for computing approximate matrix square root. The third post-compensation layer is set to counteract the adverse effect by multiplying the trace of the square root of the covariance matrix. In this work, we erase the subsequent ConvNet layers of FAST-MPN-COV and directly take the outputting symmetric matrix of the meta-layer as a $c_1(c_1+1)/2$ dimensional vector F_2 as shown in equation (2):

$$F_2 = H_{Fast}\left(\tilde{x}, \tilde{y}, \left\{H_{resnet,} \sum_{cova,} Y_{ns,} Z_{ns}\right\}\right) \tag{2}$$

where the function H_{Fast} represents multiple iterative layers of the FAST-MPN-COV with the inputs $(\tilde{x}, \tilde{y})$ and the transferring paper metres H_{resnet} of ResNet50. $\sum_{cova}$ denotes the covariance matrix of the output of the last convolution layer. Based on the pre-normalisation of $\sum_{cova}$ by trace or Frobenius norm, Y_{ns} and Z_{ns} are intermediate variables of Newton-Schulz iteration, which are suitable for parallel implementation on GPU, deriving the corresponding gradients of back propagation. Hence, with both architectures, the covariance matrix $\sum_{cova}$ is of size 256 × 256 and F_2 outputs a 32,896-dimensional vector as the image representation.

Subsequently, we describe the NTS-Net model, which consists of bilinear attention pooling, weakly supervised attention learning, and post-processing, to complete the proposed overall network structure for the fine-grained classification and object localisation. The NTS-Net applies the ResNet neural network backbone to generate feature maps F and attention maps in two-steam parallel structure size by one or several convolutional operations from input image batches. Attention maps a_{k} are then split into M maps, reflecting the region of k-th object's part. After that, feature maps F_v are element-wise multiplied by each attention map a_{k} with the same size to generate M part feature maps, which are then injected into additional local feature extraction function $g()$ to extract discriminative local feature representation. The final part feature matrix F_4 is concatenated by concatenating these local features with the bilinear attention pooling Γ, which can be represented by equation (3):

$$F_4 = \Gamma\left(\Pi_{k-1}^{M} g\left(a_{\mathrm{k}} \odot F_v\left\{\tilde{x}, \tilde{y}, H_{resnet}\right\}\right)\right) \tag{3}$$

where H_{resnet} presents the hyper-parameters of ResNet network. $\odot$ indicates element-wise multiplication for two feature tensors. In the following experiments, $g()$ is set as the global average pooling operation. During training, the initial learning rate is set to 0.001, with exponential decay of 0.85 after every five epochs. The weight of attention regularisation is set to 1.0 and the attention dropout factor is set to 80%. Then, we obtain

the local feature map vector F_4 performed on $c_2 \times w_2 \times h_2$ dimensionality, where the size of feature map is $w_2 \times h_2 = 14 \times 14$ and the channel number is $c_2 = N^M$ with $N = 512$ and $M = 7$.

Among them, we revisit some popular ResNet model tweaks. One basic tweak is replacing the 7 × 7 convolution in the input stem with five conservative 3 × 3 convolutions, which lower the computational cost and permit the input of augmented datum with a larger 448 × 448 size. Then, we introduce the Inception-v4 module with residual connections module as the similar implementations of [Inception-v4, Inception-ResNet, and the Impact of Residual Connections on Learning] and adopt batch normalisation module right after each convolution and before activation to improve the single-frame recognition performance. Finally, we abandon the final average pooling layer, the 1000-d full convolution layer and the softmax layer to extract the feature map vector F_1 and F_3 as shown in equation (4).

$$F_{1,3} = H_{resnet}\left(\tilde{x}, \tilde{y}, \{W_3, b_3, \delta\}\right) \tag{4}$$

where the function H_{resnet} can represent multiple convolutional layers of the ResNet architecture with the inputs $(\tilde{x}, \tilde{y})$ denoted to the first of these layers. W_3 denotes a square weight matrix asymptotically approximating complicated combination of multiple layers. b_3 can perform the biases of linear projection by the shortcut connections to match the dimensions, channel by channel. And δ denotes the nonlinear activation functions, which was selected as ReLU. We initialise the weights as in [Bag of Tricks for Image Classification with Convolutional Neural Networks] and train all plain/residual nets from scratch with a weight decay of 0.0001 and a momentum of 0.9. The learning rate starts from 0.1 and is divided by 10 when the error plateaus. Then, we obtain the feature map vector F_1 performed on $N \times c_3 \times w_3 \times h_3$ dimensionality, N representing the mini-batch size, c_1 representing the channel number of feature map, w_3 and h_3 denoting the width and height of each map. Thus, we use SGD with a mini-batch size of 512, and feature map is converted to the size as $w_3 \times h_3 = 7 \times 7$ and the channel number as $c_3 = 2048$.

In addition to the proposed multi-stream structure, we also propose pre-training strategy to learn professional domain knowledge from the large-scale data set. We initially pre-train our ResNet50 and VGG19 models on the ImageNet 2012 classification data set that consists of 1000 classes with the 1.28 million training images and the 50k validation images. In this way, the network learns the common classification information and acquires domain knowledge during the pre-training process, and masters the fine-grained discriminative information during the fine-tuning process. This strategy enables the network to learn the features of the target data set accurately and comprehensively, which can effectively improve the representation performance of neural networks on fine-grained small-scale data sets.

3.3 NetVLAD aggregation

The separability of features is the premise of the algorithm, if the extracted features are inseparable, it is meaningless to conduct network training blindly. Therefore, after feature extraction, this paper uses NetVLAD (Arandjelovic et al., 2016) (Vector of

Locally Aggregated Descriptors) for feature aggregation to better classify. Vector of locally aggregated descriptors proposed is a popular descriptor pooling method for both image recognition and classification. VLAD is a vector that can be used to aggregate local features (such as SIFT, SURF). And its function is similar to the Fisher vector while it is easier to get the feature space. VLAD captures the local descriptors' statistics aggregated over the image and stores the sum of residuals (difference vector between the descriptor and its corresponding cluster centre) for each visual word. The algorithm steps of the VLAD are as follows:

Using the traditional methods or the popular deep learning framework mentioned above to extract the local features, where each local descriptor is represented by x. Secondly, a codebook $C = \{c_1, \ldots, c_k\}$ of k visual words should be learned with *k*-means. And then, the features of each image are quantised and each local feature is aggregated in the nearest cluster centre. After quantifying, the feature space is divided into several subspaces called cells. Next, accumulate the differences $x - c_i$ of the vectors x assigned to c_i. Finally, the descriptor could be represent by $v_{i,j}$, where the indices i and j, respectively represent the cluster centre and the local descriptor component. And indices l represent the number of the local vector x, $a_i(x_l)$ denotes the membership of the x_l descriptor to i-th visual word. We encode it as 1 or otherwise set to 0 when cluster c_i is the cluster closest to the descriptor x_l. Assuming that there are N-dimension descriptors, a component of v is obtained as a sum over all the image descriptors:

$$v_{i,j} = \sum_{l=1}^{N} a_i(x_l)\left(x_l(j) - c_i(j)\right), i = 1 \ldots k, j = 1 \ldots d \tag{5}$$

where x_j and $c_i(j)$ respectively denote the *j*-th component of the descriptor x considered and of its corresponding visual word c_i, then the vector v is subsequently L2-normalised by $v = v / \| v \|_2$.

On the basis of the above, the NetVLAD strategy is used to training the aggregating pooling layer in the CNN framework, the centre of the NetVLAD could be adjust during the training process and need not be located at the centre of the cell, which can reduce the residual between clusters c_i and x_l to get a compact feature. With this global feature, NetVLAD can better describe the entire image. In order to make this layer trainable, this paper rewrites the hard assignment $a_i(x_l)$ to be a soft assignment as equation (6).

$$\overline{a}_i(x_l) = \frac{e^{-\alpha \| x_l - c_i \|^2}}{\sum_{i'} e^{-\alpha \| x_l - c_{i'} \|^2}} \tag{6}$$

where α takes a constant value between 0 and 1, which makes the weights of descriptor x_l to cluster c_i proportional to their proximity. The weight of the descriptor increases as the distance from the cluster centre decreases. Equation (7) can be obtained by simplifying equation (6):

$$\overline{a}_i(x_l) = \frac{e^{w_i^T x_l + b_i}}{\sum_{i'} e^{w_{i'}^T x_l + b_{i'}}} \tag{7}$$

The parameters w_i, b_i, c_i can be updated during the training stage. This makes the newly NetVLAD layer more flexible and could get a cluster centre position where the residual is further reduced. The final form of the aggregation layer is obtained by integrating the soft assignment into the VLAD descriptor equation (5) as shown following:

$$V(j,i)=\sum_{l=1}^{N}\frac{e^{w_i^T x_l+b_i}}{\sum_{i'} e^{w_{i'}^T x_l+b_{i'}}}\left(x_l(j)-c_i(j)\right) \tag{8}$$

Similarly to the original VLAD descriptor, the NetVLAD layer aggregates the first order statistics of residuals $\left(x_l(j)-c_i(j)\right)$ in different parts of the descriptor space weighted by the soft-assignment $\overline{a}_i(x_l)$ of descriptor x_l to cluster c_i. Note however, that the NetVLAD layer has three independent sets of parameters w_i, b_i, c_i, compared to c_i of the original VLAD, which enables greater flexibility than the original VLAD.

With aforementioned operations, our proposed fusion model gain an overall representation of prediction score in decision-level perspective, which actually is the joint posterior probabilities by integrating several prior probabilities from each component model of Multi-steam DNN feature extractor. Finally, we add two full convolution layer, a dropout layer and a softmax layer after the VLAD aggregation module to output the normalised classification result. As the softmax operator obtains predicted probabilities, the cross entropy loss is used to estimate the degree of inconsistency between the predicted score and the true label. The optimal solution of loss is to minimise the error gap to small enough value with some regularisation constraints including L1 or L2 terms. It encourages the output scores dramatically distinctive, which potentially leads to overfitting for intra-class description. This easily leads to the low inter-class recognition in dealing with other categories. Thus, during training we optimise the following multi-part loss function:

$$Loss=L_{agg}+\lambda_1 L_{feature}=-\sum_{c=1}^{W}\tilde{y}_c\log(S_c)-\lambda_1\sum_{i=1}^{2}\sum_{c=1}^{W}\tilde{y}_c softmax(F_i) \tag{9}$$

where $L_{feature}$ indicates the loss of multi-DNNs feature extractor, and L_{fuse} denotes the partial loss of aggregation module. Moreover, we introduce the weighting factors $\lambda_1 \in [0, 0.5]$ to balances the importance of each loss. Our proposed loss is a simple extension to softmax that we consider as an experimental baseline to differentiate inter-class discrepancy among fine-grained categories. Specifically, W indicates the number of categories. $\tilde{y}_c$ indicates the indicator variable (0 or 1) if the category and sample have the same category, otherwise 0. S_c denotes the predicted probability that the observed sample belongs to category c. Similarly, we obtain the whole loss of three models extracting various feature maps F_i with the softmax function. In subsequent experimental results, we use the above loss form to optimise the entire model structure, which is demonstrated to be effective in improving the performance for fine-grained visual classification tasks.

4 Results and discussion

This section compares the performance of state-of-the-art deep-learning models, fine-grained models, and their fusion model on the data sets created in this article. The results are presented in the classification. To ensure the reliability of training, this paper divides 10% of the data set into a test set, a total of 11,433 pictures, the rest as a training and verification set. The experiments with seven deep-learning classification architectures – including SENet (Hu et al., 2018), VGG19, ResNet50, Densenet121, and fine-grained classification models such as NFL (Wang et al., 2018), NTS-Net, FAST-MPN-COV were carried out. Finally, this paper also applies decision-level fusion of these fine-grained models – MSA-NET on the data set. All models were and tested on an Intel Core i7 3.6 GHz processor with four NVIDIA Tesla p40 GPUs and 256 G RAM. As shown in Table 1, the accuracy of the classification is obtained for fusion model MSA-NET, and the accuracy of the current classification model on a data set.

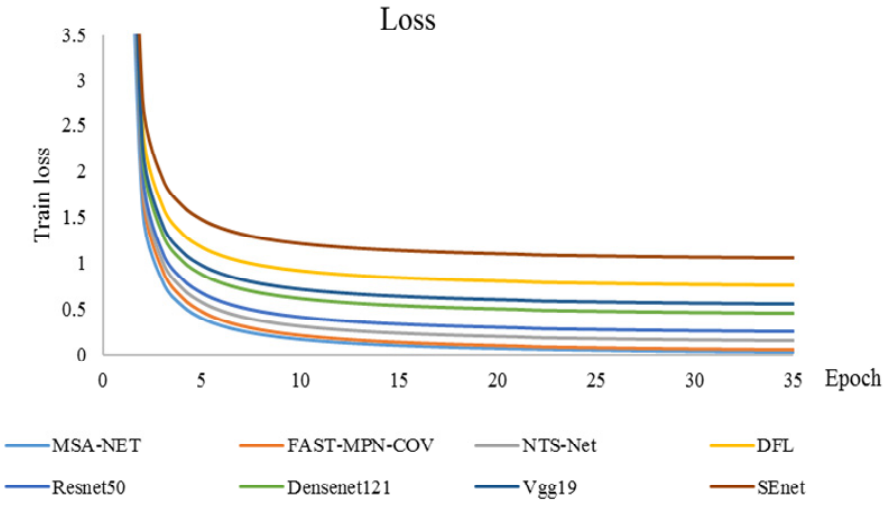
Table 1 Experimental results "Acc" denotes the top-1 accuracy in percentage "time" denotes the time spent on the test

Method	*Accuracy (%)*	*Time(s)*
VGG19	78.14	440
ResNet50	85.74	405
Densenet121	81.38	380
SENet	81.38	380
NTS-Net	86.67	138
FAST-MPN-COV	87.58	729
DFL	75.81	734
MSA-NET	91.18	756

As shown in Table 1, the current popular deep learning models have a certain accuracy in classification; for example, ResNet50 achieves an accuracy of 85.74%; however, the fine-grained model MSA-NET can achieve better results (such as FAST-MPN-COV increased by 1.84%, NTS-Net increased by 0.93%). On this basis, the fusion of these fine-grained models achieves a higher recognition rate. Overall, the accuracy of the model after fusion is higher than the accuracy of a single model prior to fusion. Explain that the fusion model can solve the fine-grained image classification problem well.

At the same time, this paper also observes the training loss of these models, as shown in Figure 4. Due to the complexity and largeness of the data set, compared with fine-grained classification network, some popular classifiers such as VGG19 and Densenet-121 are obviously not as effective as fine-grained classification model training. What's more, on this basis, the model after fusion has a better training effect than all models including single fine-grained models. To a certain extent, this training loss also proves that the fused fine-grained model has a better effect on the identification of pests and diseases.

Figure 4 Training loss (see online version for colours)



Owing the complexity of the patterns shown in each class, especially in terms of form and background, the system tends to be confused on several classes, which results in lower performance. Figure 5 presents the confusion matrix of the final recognition results. Based on the results, this paper can visually evaluate the performance of the fusion model classifier and determine what classes and features are more highlighted by the neurons in the network. Further, it helps to analyse a further procedure in order to avoid that inter-class confusion. From Figure 5, the colour bar of the diagonal reflects the correct degree of classification of each category, the deeper the yellow, the higher the accuracy of classification. Based on this criterion, it can be seen that the fusion model has higher accurate classification accuracy.

Based on the results obtained by the fusion model, this paper compares the results of ResNet50, NTS-Net, Fast-MPN-COV, and MSA-NET; and quantitative accuracy analysis of each type of each model. It can be found that the same model has different accuracy for different categories. Although the individual models have different recognition capabilities for different categories, it can be seen from the results of the fusion model that the fusion model combines and complements different feature extraction capabilities of different images. The accuracy of different models in the same category is different. For example, in category 2, the accuracy of NTS-net is 84.6%, and the accuracy of Fast-MPN-COV is 76.9%. The difference in accuracy between them is relatively large, so this paper gets the results after the model fusion. The accuracy of the fusion model in the second category was 92.3%, which was 7.7% higher than NTS-net and 15.4% higher than Fast-MPN-COV. Overall, it can be found that the accuracy of each type of fusion has different degrees of improvement than the accuracy of each type before fusion so that the overall accuracy after fusion is improved. Further analysis is shown in Figure 6. For the 15 pictures in the 181 class, the red representative model predicts the wrong picture, and the rest represent the predicted image. The results obtained by the fusion model show that the results predicted by the fusion model are all

correct. It shows that the fusion model can fuse the feature extraction capabilities of NTS-Net and Fast-MPN-COV models, thus improving the accuracy of model classification.

Figure 5 Confusion matrix of the fusion model

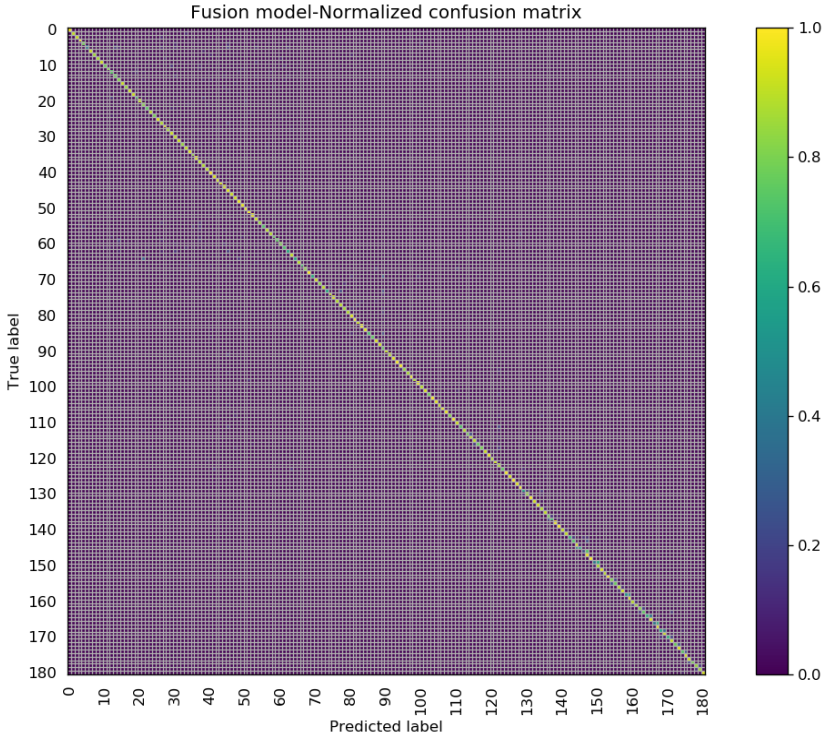
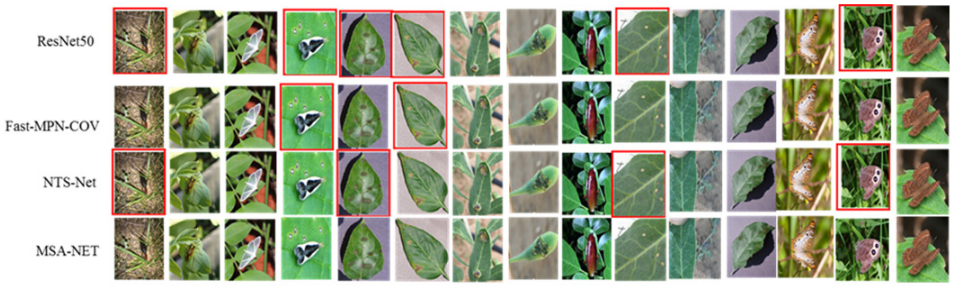


Figure 6 Model classification results on the part of the data set



5 Conclusion

In various precision agricultural tasks, scientific studies on species identification of pests and diseases have been considered as one of the most important applications. Since pests

and diseases recognition suffers the low inter-class discrepancy and high intra-class variances from the sub-categories, the fine-grained visual classification of pests and diseases is still challenging for traditional Deep Neural Networks (DNNs). In this investigation, we present a domain-specific deep-learning classification model according to practical agricultural tasks to classify 181 categories with 124,437 pieces collected by different cameras and equipment of the Internet of Things (IoT). Firstly, the proposed method employs data augmentation tricks to enlarge the data set and pretrains ResNet networks on high-quality images data sets to learn fine-tuning skills. Then, refined multiple DNNs consisting of ResNet, NTS-Net, and FAST-MPN-COV are applied to design a multi-stream feature extractor, which utilises the mixture-granularity information to exploit features distinguishing interclass and intra-class variances. Finally, a fusion module equipped with NetVLAD aggregation layer is developed to fuse different components model. Experiments demonstrate the effectiveness of the MSA-NET with higher accuracy of 91.8% at a moderate speed, which outperforms state-of-the-art deep-learning methods. In the future, we plan to add more images and annotations of new pests and disease species for fine-grained classes that are challenging to annotate. Further research is also necessary to lightweight network with model parameters compression while boosting speed and improving accuracy.

Acknowledgements

The research presented in this paper has been supported by the National Key Research and Development Program of China No. 2017YFC1600605, Beijing Municipal Education Commission Nos. KM201910011010, KM201810011005 and National Natural Science Foundation of China Nos. 61673002 and 61903009.

References

Arandjelovic, R., Gronat, P. and Torii, A. et al. (2016) 'Netvlad: cnnarchitecture for weakly supervised place recognition', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.5297–5307.

Branson, S., Horn, G.V. and Belongie, S. et al. (2014) 'Bird species categorization using pose normalized deep convolutional nets', *British Machine Vision Conference*, *arXiv:1406.2952*.

Han, J., Zhang, D. and Cheng, G. et al. (2015) 'Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning', *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 53, No. 6, pp.3325–3337.

He, K., Zhang, X. and Ren, S. et al. (2016) 'Deep residual learning for image recognition', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.770–778.

Hu, J., Shen, L. and Sun, G. (2018) 'Squeeze-and-excitation networks', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.7132–7141.

Huang, G., Liu, Z. and Van Der Maaten, L. et al. (2017) 'Densely connected convolutional networks', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.4700–4708.

Jaderberg, M., Simonyan, K. and Zisserman, A. et al. (2015) 'Spatial transformer networks', *Advances in Neural Information Processing Systems*, pp.2017–2025.

Li, P., Xie, J. and Wang, Q. et al. (2018) 'Towards faster training of global covariance pooling networks by iterative matrix square root normalization', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.947–955.

Lin, T.Y., Roychowdhury, A. and Maji, S. (2015) 'Bilinear CNN models for fine-grained visual recognition', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp.1449–1457.

Peng, Y., He, X. and Zhao, J. (2017) 'Object-part attention model for fine-grained image classification', *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*, Vol. 99, pp.1–1.

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

Wang, Y., Morariu, V.I. and Davis, L.S. (2018) 'Learning a discriminative filter bank within a CNN for fine-grained recognition', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.4148–4157.

Xiao, T., Xu, Y. and Yang, K. et al. (2015) 'The application of two-level attention models in deep convolutional neural network for fine-grained image classification', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp.842–850.

Yang, Z., Luo, T. and Wang, D. et al. (2018) 'Learning to navigate for fine-grained classification', *Proceedings of the European Conference on Computer Vision (ECCV)*, pp.420–435.

Zhang, H., Cisse, M. and Dauphin, Y.N. et al. (2017) 'mixup: beyond empirical risk minimization', *arXiv preprint arXiv:1710.09412*.

Zhang, N., Donahue, J. and Girshick, R. et al. (2014) 'Part-based R-CNNs for fine-grained category detection', *arXiv:1407.3867*.