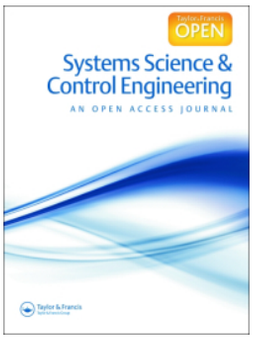


Taylor & Francis
Taylor & Francis Group

Systems Science & Control Engineering

An Open Access Journal

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tssc20

Crop pest recognition based on a modified capsule network

Shanwen Zhang, Rongzhi Jing & Xiaoli Shi

To cite this article: Shanwen Zhang, Rongzhi Jing & Xiaoli Shi (2022) Crop pest recognition based on a modified capsule network, Systems Science & Control Engineering, 10:1, 552-561, DOI: 10.1080/21642583.2022.2074168

To link to this article: https://doi.org/10.1080/21642583.2022.2074168

Published online: 17 May 2022.

SYSTEMS SCIENCE & CONTROL ENGINEERING: AN OPEN ACCESS JOURNAL
2022, VOL. 10, NO. 1, 552–561
https://doi.org/10.1080/21642583.2022.2074168

Crop pest recognition based on a modified capsule network

Shanwen Zhang, Rongzhi Jing and Xiaoli Shi

School of Electron Information Engineering, Zhengzhou SIAS University, Zhengzhou, People's Republic of China

ABSTRACT

Crop pest insects seriously affect yield and quality of crops, and pesticide control methods cause severe environmental pollution, which has inextricably influenced people's daily lives. Crop pest identification in the field is crucial components of pest control. It is much more complex than generic object recognition due to the apparent differences in the same pest species in the field with various shapes, colours, sizes and complex background. A crop pest recognition method is proposed based on a modified capsule network (MCapsNet). In MCapsNet, a capsule network is used to improve the traditional convolutional neural network (CNN), and an attention module is introduced to capture the most important classification features and speed up the network training. The experimental results on a pest image dataset validate that the proposed method is effective and feasible in classifying various types of insects in field crops and can be implemented in the agriculture sector for crop protection.

ARTICLE HISTORY
Received 10 December 2021
Accepted 2 May 2022

KEYWORDS
Crop pest detection; capsule networks (CapsNet); attention mechanism; modified CapsNet (MCapsNet)

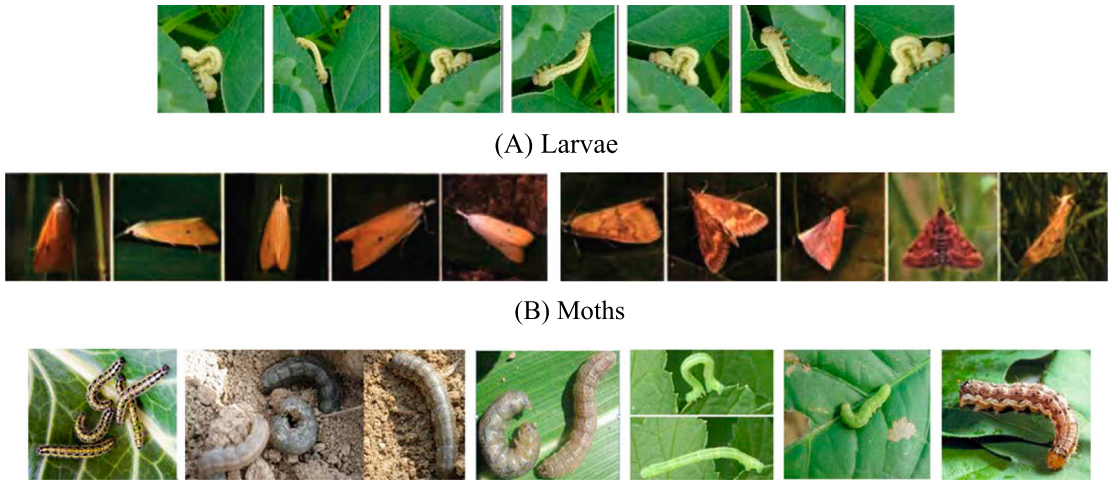
1. Introduction

Crops, such as maize, wheat, rice, soybean, sugarcane and cotton, are often affected by different kinds of pests, which severely reduce the production and quality of crops. Accurate detection and identification of crop pests is taken care of pest control in the earlier stage of crop growth. Its detection and recognition of crop insects remain difficult and challenging on a massive scale due to their similar appearance and complex background. Moreover, the pest images collected in the natural field environment are often affected by illumination, insect morphology, image size and shooting angle, etc., which greatly complicate pest detection and recognition (Wu et al., 2019b). Using the continuous development of image processing and pattern recognition technology, many crop pest detection and identification methods have been presented (Deng et al., 2020; Zhang et al., 2013), which can be divided feature extraction-based methods (Miranda et al., 2014) and deep learning-based methods (Thenmozhi & Reddy, 2019).

Feature extraction and the classifier are crucial of pest recognition in the field. Wang et al. (2012a) designed an automatic identification system to identify insect images, while several relative features are extracted by digital image progressing, pattern recognition and the theory of taxonomy, and artificial neural networks (ANNs) and support vector machine (SVM) are used as classifiers for pest identification tests. To extract the invariant features for representing the pest appearance, Deng et al. (2018) integrated the bio-inspired hierarchical model, scale invariant feature transform (SIFT), and non-negative sparse coding (NNSC) to increase the feature invariance and then extracted the invariant texture features based on a local configuration pattern algorithm. Zhang et al. (2014) presented a pest image recognition method by combining the features of colour, shape and texture and validated its effectiveness using 34 kinds of pest images, including rice, rape, corn and soybean. Wang et al. (2012b) conducted a lot of tests on eight and nine orders with different features, compared the advantages and disadvantages of their system, and designed an automatic insect identification system at the order level according to the methods of image processing, pattern recognition and the theory of taxonomy and provided some advice for future research on insect image recognition.

Although the various image segmentation algorithms (Wang et al., 2010a; Wang & Huang, 2008), feature extraction algorithms (Chen et al., 2005; Du et al., 2006b; Huang & Jiang, 2012; Wang & Huang, 2009), neural networks (NNs) (Han et al., 2008, 2010; Han & Huang, 2006; Zhao et al., 2004) can be applied the pest image recognition task (Deng et al., 2018; Miranda et al., 2014; Zhang et al., 2013), they carry on the tedious steps, such as image preprocessing, image segmentation, feature extraction and feature classification, to recognize the crop pests, and the corresponding results rely heavily on the effect of

CONTACT Xiaoli Shi 844394899@qq.com



(A) Larvae

(B) Moths

(C) Some kinds of crop pests with various poses, colours, shapes, scales and rotations

Figure 1. Crop pest images collected in nature fields.

the image preprocessing and segmentation (Du et al., 2007; Shang et al., 2006; Xiao-Feng et al., 2008a), extracted handcraft-features and classifiers, such as NNs (Huang & Du, 2008), radial basis probabilistic NNs (Du et al., 2006a; Huang, 1999) and background (Kamilaris & Prenafeta-Boldú, 2018). Moreover, because the same pest in the field is different and irregular with a various shapes, poses and colours and complex backgrounds, as shown in Figure 1, using the features designed manually is difficult to obtain the feature expression closest to the natural attribute of the crop pests in the field, then it is difficult to extract the robust and invariant classification features from the pest images.

Image recognition using deep learning is considered the state of the art in computer vision research (Zhang et al., 2018). Deep learning-based crop pest detection methods not only can save time and effort, but also can achieve a real-time judgment. It can perform automatic feature extraction and learn complex high-level features in image classification applications (Shah et al., 2020). Due to the ability to learn the data-dependent features automatically from the data, many convolutional neural networks (CNN) and their variant models have been applied to pest identification tasks (Liu & Wang, 2020). Cheng et al. (2017) built an image dataset of tomato diseases and pests under the real natural environment, and proposed a tomato diseases and pests detection method based on improved Yolo V3. It references intelligent recognition and engineering application of plant diseases and pests detection. To achieve pest identification with the complex farmland background, Alfarisy et al. (2018) proposed a pest identification method using deep residual learning. It can be integrated with current agricultural networking systems into actual agricultural pest control tasks. Wu et al. (2019a) collected 4511 images of paddy pests and diseases from four languages using search engines and augmented them to develop diverse datasets. These datasets were fed into the CaffeNet model and processed with the Caffe framework for paddy pests and diseases recognition and achieved an accuracy of 87% higher than the random selection of 7.6%. To control crop diseases and pests, Nanni et al. (2020) introduced AlexNet and GoogleNet to detect crop pests and diseases on several crop pest and disease images. They achieved the highest detection accuracy rate of 98.48% for 38 pests and diseases with a higher efficiency, practicability, and accuracy. Li et al. (2020b) proposed an automatic classifier for pest recognition by integrating saliency methods and CNNs, where three different saliency methods are used as image preprocessing to create three images for every saliency method, and some new 3×3 images are created for every original image to train different CNNs. Thenmozhi & Reddy (2019) introduced a crop pest recognition method based on several deep CNNs that accurately recognize ten common crop pests. Chen et al. (2021) proposed a convolutional neural network (CNN) model to classify insect species on three publicly available insect datasets. The proposed model was evaluated and compared with pre-trained deep learning architectures such as AlexNet, ResNet, GoogLeNet and VGGNet for insect classification. The experiment results validated that CNN can comprehensively extract multifaceted insect features. To enhance the learning capability for pest images with cluttered backgrounds, Li et al. (2020a) proposed a crop pest recognition by the attention-embedded lightweight network under field conditions, where the optimized loss function and two-stage transfer learning are adopted in model training to

improve the identification accuracy of crop pest images. Wang et al. (2020) established a large-scale standardized agricultural pest dataset, Pest24, containing 25,378 labelled images of 24 kinds of field pests. They applied several state-of-the-art deep learning detection methods, Faster RCNN, SSD, YOLOv3, and Cascade R-CNN, to detect the pests in the dataset, and obtain encouraging results for real-time monitoring field crop pests, analyzed the datasets in a variety of aspects, finding that relative scale, number of instances and object adhesion, mainly influence the pest detection performance, and conducted crop pest detection experiments using deep learning networks.

The modelling ability and classification performance of CNN and its improved models for geometric invariant features mainly come from the expansion of datasets, the deepening of network layers and the artificial design of the model, but it does not fundamentally solve the deformation problem of field pests (Chen et al., 2021; Li et al., 2020a, 2020b; Wang et al., 2020).

CNN and its improved models perform very well in image classification and recognition, but they do not fundamentally solve the deformation problem of field pests, because the image colour of the field pest is rich, its colour gradient change is significant, and its size and shape characteristics are various and complex, using CNN directly cannot extract the depth rich characteristics of the pest image. Unlike the maximum pooling in CNN, Capsule Network (CapsNet) does not discard the location information between entities within the region and it retains semantic information and spatial relationships between the various features in the text classification (Kumar et al., 2020). A capsule is composed of neurons, where the texture, colour and other characteristic attributes in the input image are contained in the neurons. It can predict the global characteristics of the whole entity through some attributes of the entity in the neurons. CapsNet is ideal for the segmentation and recognition of handwritten and medical images. It can be applied to crop recognition tasks.

Capsule of CapsNet is composed of neurons, where the texture, colour and other characteristic attributes in the input image are contained in the neurons. It can predict the global characteristics of the whole entity through some attributes of the entity in the neurons. It is a carrier of multiple neurons to identify a limited observation condition and deformation within the scope of the visual entities, and the output of a set of the instantiation of the parameters and the degree of its significant value (that is, the probability of the entity), the parameters include the entity's precise location, colour information, and the shape of the information. Among them, the significant degree of the existence of visual entities is locally invariant (the recognition probability of the entity type does not change), while the instantiation parameters are isotropic (the instantiation parameters also change correspondingly). In CapsNet, the neural nodes in CNN are replaced with the neuron vector, and the new neural network is trained using the dynamic routing algorithm (DRA) instead of the maximum pooling operation in CNN. It only considers the characteristics of image pixels when extracting image features and fully considers the spatial relations of image elements. The convolutional layer and Primary Capsule layer is the main feature extraction part of achieving the matching and projection from the image's low-dimensional features to high-dimensional features. During the whole capsule operation, the lower capsule transmits part of the extracted features to the upper capsule for overall recognition. DRA is the key to feature information projection between capsules in CapsNet (Zhang et al., 2020). CapsNet is ideal for the segmentation and recognition of handwritten and medical images (Sabour et al., 2017). But because the image colour of the field pest is rich, its colour gradient change is significant, and its size and shape characteristics are various and complex, using CapsNet directly cannot extract the depth rich characteristics of the pest image (Kromm & Rohr, 2020). Inspired by CNN, CapsNet and attention mechanism, a modified CapsNet with attention mechanism, namely MCapsNet, is constructed for pest recognition. The main contributions of this paper are given as follows.

(1) MCapsNet is constructed to extract the invariant features from the various pest images.
(2) The attention mechanism is used to capture rich contextual relationships for better feature extraction and improving network training.
(3) A LeakyReLU activation function is used to speed up the model convergence to some extent and to prevent gradient dispersion.

The remainder of this paper is organized as follows. Section 2 introduces the attention mechanism and CapsNet. Section 3 presents an improved CapsNet with an attention mechanism in detail. The experiments and results are presented in Section 4. Section 5 summarizes this paper and gives future works.

2. Related works

2.1. Attention mechanism

Attention mechanisms have been successfully applied to the deep learning-based recognition tasks, such as machine translations, question answering, speech

recognition and image captioning (Li et al., 2020a). Its central idea is to assign a high weight to the important features and ignores irrelevant features, and then amplify the desired features. Let an input matrix $H = [h_1, h_2, \ldots, h_n]$, where h_i is the i-th output feature vector, and n is the length of a sentence. The output weight component $\alpha_i \in R^n$ of H is obtained by tan h function,

$$\alpha_i = \tan h(W_\alpha H + b) \tag{1}$$

where $W_\alpha \in R^{1\times n}$ is a weight matrix, and b is a bias.

The attention weight $\beta_i \in R^n$ is calculated by the Softmax function,

$$\beta_i = \exp(\alpha_i) / \sum_{j=1}^{n} \exp(\alpha_j) \tag{2}$$

The output feature is obtained by

$$\gamma = \sum_{i=1}^{n} \beta_i H \tag{3}$$

2.2. Capsule network (CapsNet)

CapsNet is mainly composed of a convolutional layer, primary capsule layer (PrimaryCaps), digital capsule layer (DigitCaps) and concatenation layer, as shown in Figure 2.

Suppose the input data is an image of 28 × 28 pixels. The convolutional layer aims to capture the features from the input image and output the feature graphs, which are transformed into vector capsules by the PrimaryCaps layer, then are output after the calculation of Digitalcaps. Digitalcaps is similar to the full-connection layer of CNN, but each neuron is transformed into a capsule structure for classification and output. The probability of a certain category is measured by the size of the output vector mode, and the vector with the largest mode is the output category. In the classical CapsNet, the convolution layer selects the convolution kernel with a step size of 1, depth of 32 and 9 × 9. ReLU is selected as the activation function. In the PrimaryCaps layer, eight groups of convolution kernels with a step size of 2, depth of 32 and size of 9 × 9 are selected, and eight times of convolution operations are carried out on the feature images output by the convolution layer to obtain 6 × 6 × 8 × 32 feature vectors, and 1152 capsules are obtained by the feature vectors. Each capsule consists of an eight-dimensional vector. DigitCaps layer outputs tensors of 16 × 10. The algorithm of CapsNet is shown in Figure 3.

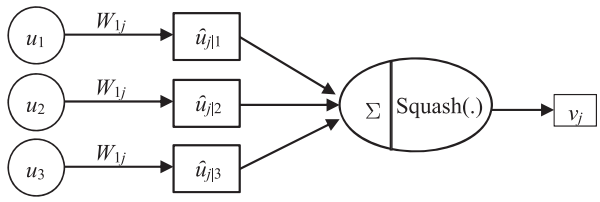


Figure 3. Computation of CapsNet.

In Figure 3, W is the back-propagation operation in parameter update, and C is the weighted sum operation of scalar. The right of Figure 3 is the capsule network part; DRA iterations can get better classification results when set to 3, so in this model, the dynamic routing number of iterations of three PrimaryCaps convolution kernel size is set to 3 × 3, the channel number is set to 3, the number of Digitalcaps tag set to data set on the number of the categories. The feature vector is input into the capsules. The feature mapping relationship between low-level capsules and high-level capsules is encoded in the weight matrix W, and the feature vector is multiplied by the corresponding weight matrix W. To better concatenate the feature vectors in the previous low-level capsules, the prediction vectors are weighted sum of the scalar before input into the high-level capsules. The capsule network adopts the Squashing nonlinear function to ensure that the length of the output vector is between 0 and 1. The j-th output capsule of the parent capsule layer is calculated as

$$v_j = \frac{||s_j||^2}{1 + ||s_j||^2} \cdot \frac{s_j}{||s_j||} \tag{4}$$

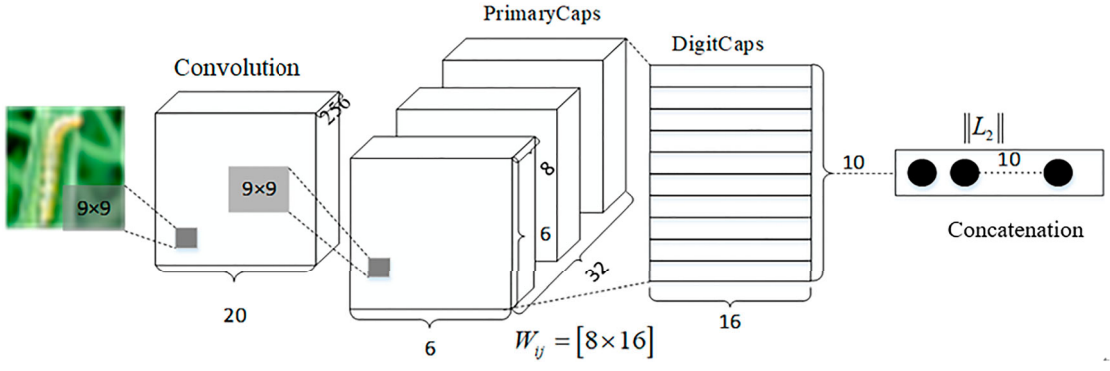


Figure 2. The architecture of CapsNet.

where s_j is the total input vector of the j-th capsule obtained by the weighted sum of the j-th parent capsule layer connecting with the i-th child capsule layer, $||s_j||^2/(1+||s_j||^2)$ is the reduction coefficient of s_j, $s_j/||s_j||$ is the normalized unit vector of s_j, $s_j = \sum_i c_{ij}\hat{u}_{j|i}$, the prediction vector $\hat{u}_{j|i}$ is obtained by multiplying the output features of the BN layer with the weight matrix of the primary capsule layer, and the prediction vector is obtained as follows:

$$\hat{u}_{j|i} = w_{ij}u_i \tag{5}$$

where u_i is the i-th output capsule of the child capsule layer, and w_{ij} is the elements of the weight matrix W.

Similar to the fully connected neural network in CNN, s_j is the linear weighted sum of u_i of the upper CapsNet. On this basis, v_j is introduced into CapsNet. The coefficient c_{ij} is updated by DRA between capsules, as follows.

$$c_{ij} = \exp(b_{ij})/\sum\nolimits_k \exp(b_{ij}), b_{ij} + \hat{u}_{j|i}.v_j \rightarrow b_{ij} \tag{6}$$

where k is the category number.

CapsNet uses the edge loss function as the model loss function, which is expressed as follows.

$$L_c = \sum_{k\in \mathrm{CNum}} T_k \max(0, m^+ - ||V_k||^2) + \lambda(1-T_k)\max(0, ||V_k|| - m^-)^2 \tag{7}$$

where the former part is used to calculate the settings of the correctly classified digital capsule, and the latter part is used to calculate the losses of wrongly classified digital capsules, $m^+ = 0.9$ and $m^- = 0.1$ are the default category prediction values, $\lambda = 0.5$ is the default balance coefficient, T_k is the label of data category, $T_k = 1$ is the correct label, CNum is the number of categories, $||V_k||$ is the length of the vector representing the probability of discriminating as the class kth pest, the total loss is the sum of all digital capsule loss functions.

In Equation (6), the smaller L_c, the smaller the difference between the predicted value of the output vector and the true value of the input vector, that is, the better the CapsNet classification effect. To estimate the error between the predicted values and the real results to update the parameters of the model by the dynamic routing algorithm, the number of v_j is consistent with the model output of the number of categories in the last layer of the digital capsule. The length of a vector v_j is calculated for the classification probability expressing the target belonging to the jth class. The correct category and the wrong category are 1 and 0, respectively. If the judgment is correct, the first half is used to calculate the loss. If the judgment is incorrect, the latter part is used to calculate the loss.

CapsNet classifies the input features by adopting DRA instead of the pooling layer of CNN. The more similar features, the stronger the features are, which is equivalent to a feature selection process (Kumar et al., 2020; Zhang et al., 2020). The main idea of the DRA is that all the sub-capsule outputs can predict the instantiation parameters of the parent capsule through the alternating matrix. When the bottom capsule prediction is the same, the parent capsule activates and outputs the eigenvector.

3. Modified CapsNet with an attention mechanism

CapsNet is not effective in large-scale image datasets, which limits its application range. A modified CapsNet (MCapsNet) is constructed by introducing an attention mechanism and local DRA (LDRA) instead of the DRA of CapsNet. Firstly, the two-dimensional images are extracted through three convolution layers and then transmitted to the feature capsule layer to form the high-dimensional feature capsule. Then, the LDRA of the category capsule layer is mapped to the final classification result.

The convolution part of CapsNet uses only a convolution operation to extract the characteristics of the image, which cannot extract the high-level characteristics of the pest image. In MCapsNet, three continuous convolutional layers are used to replace the single convolutional layer of CapsNet to achieve high-level feature extraction of pest images. Suppose pest images with a size of $224 \times 244 \times 3$ are selected as the input of MCapsNet. After the 64-layer feature map is first generated, the size of the feature map is 112×112, and a LeakyReLU nonlinear function is introduced between the convolution layers to activate the convolution operation and then transfer to the next layer. After the same convolution operation, the feature graph generated at the previous layer is transformed into a feature graph of 128 layers with a size of 5656. The last convolutional layer continues to carry out the convolution operation on the feature graph after twice activation. Two hundred fifty-six convolution kernels with the size of 3×3 are used to carry out mobile convolution against the feature graph generated by the middle convolutional layer and are generated before activation. MCapsNet performs pre-convolution and homogeneous layer activation operations using three successive convolutions with step 2 and size 3×3. Through three continuous convolutions, the $224 \times 244 \times 3$ pest images are transformed into two-dimensional image features, which are conducive to feature analysis and processing of capsule layer to better abstract the analysis of extracted features and enhance the expression of two-dimensional image features by feature capsule layer.

To avoid the influence of gradient disappearance in the back propagation of pre-convolution on the classification results of pest images, an appropriate activation function is used to sort out the features propagated at the next layer. To make the model have better fitting performance and convergence speed, the activation function in CapsNet is replaced, and each convolution layer in the network with pre-convolution is activated. The nonlinear activation function is added to activate the image features and remove some redundant image features, and MCapsNet adopts the LeakyReLU activation function to speed up model convergence to some extent and prevent gradient dispersion. The difference between LeakyReLU and ReLU is that ReLU will go to 0 when the input X is less than 0, while LeakyReLU retains some information and the gradient is not zero. LeakyRELU is defined as follows

$$\text{LeakReLU}(x) = \begin{cases} x, x > 0 \\ \text{leak} \cdot x_i, x \leq 0 \end{cases} \tag{8}$$

where the leak is a decimal and the value is 0.1, x is the input data.

To adapt to different objective functions, MCapsNet adopts the Adam algorithm of adaptive learning rate. Its gradient is smoother in the training process, and all parameters are optimized. Its adaptive learning rate is based on adaptive low-order moments to accelerate the convergence of MCapsNet. For pest images with large noise, the Adam algorithm can reduce the impact of noise on feature extraction. In training, the adaptive learning rate and momentum algorithm are used, the learning rate is always kept within a fixed range, the parameter changes are relatively stable, and gradient descent can be avoided. Updating the model's weight and parameters makes MCapsNet have better performance. Making full use of the mean of the first and second moments of the matrix is an important method of updating the model. To further solve the over-fitting problem caused by too long training time in MCapsNet, the two-stage model training method is used. In the second stage, the model parameters are initialized by less training, and then the model training of all the data is carried out.

From the above analysis, an MCapsNet with an attention mechanism model is constructed. Its architecture and corresponding parameters are shown in Figure 4 and Table 1, respectively. Three continuous convolutional operations in Layer_1 of MCapsNet are shown in Figure 5.

Based on CapsNet, MCapsNet adjusts the input feature graph and the size of convolution kernel, and utilizes three successive convolutional layers, an attention mechanism layer and a convolution layer to generate a 28×28 feature map as input of PrimaryCaps layer and outputs 50,176 units. The feature maps are encapsulated in groups according to the size of 8×1 capsules by Digitalcaps operation, and finally encapsulated into 6272 capsules. After the adaptive modification of the parameters, the feature capsule layer can express the rich details of the mural image. In the attention mechanism, a self-attention module is used to focus on the significant extracted features of the three successive convolutional layers while ignoring the needless features, which is useful for extracting the crucial features of plant pest images. After the similarity output matrix is obtained, mask calculation of

Table 1. Related parameters of MCapsNet.

Layer	Type	Size	Channel	Output
Layer_1	Conv 11	3×3	64	$112 \times 112 \times 64$
	Conv 12	3×3	128	$56 \times 56 \times 128$
	Conv 13	3×3	256	$28 \times 28 \times 256$
Layer_2	Attention	–	–	–
Layer_3	Conv 31	3×3	256	$26 \times 26 \times 256$
Layer_4	Main CapsNet 41	3×3	256	$14 \times 14 \times 32 \times 16D$
Layer_5	Minor CapsNet 42	5×5	–	$8 \times 8 \times 32 \times 16D$
Layer_6	Conv 61	1×1	256	64×256
Layer_7	Concatenation			16,384

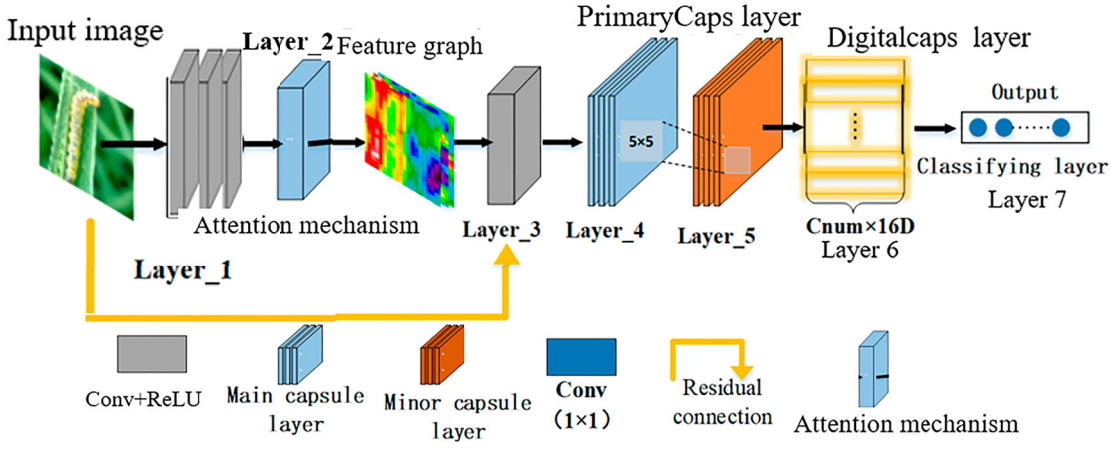


Figure 4. The architecture of MCapsNet.

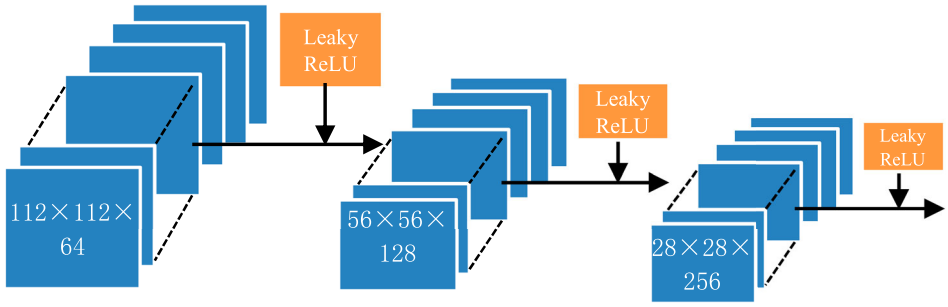


Figure 5. The first three convolutional operations in Layer_1 of MCapsNet.

global attention is carried out with the passing output. The output features from the self-attention module are transferred to the second convolution layer, and the 3×3 convolution kernel is used further to improve the feature description ability of the model. In the PrimaryCaps layer, the input characteristic graph of Conv2 is vectorized, in which the PrimaryCaps is divided into main and minor capsule layers. Each layer comprises 32 capsules, and each capsule is composed of 16 5×5 convolution nuclei. The Digitalcaps layer is composed of a Cnum $\times$ 16D dimension vector, which predicts the type of input image. Routing between capsule layers is realized by LDRA. In model training, the loss function $L_{\text{loss}}(\text{Vector}_{\text{out}}, R_r)$ is calculated, and the model parameters are updated according to the back-propagation algorithm.

In the convolutional layer, the attention mechanism is used to learn more effective feature maps to make the network pay more attention to the foreground regions. The attention block consists of a convolutional layer and a gating mechanism. In this convolutional layer, a convolution kernel $V \in R^{k \times 1}$ is applied to all the context features with window size k (padded when necessary) of every feature $c_j (j = 1, 2, \ldots, n-h+1)$ in feature map C to produce the attention weight matrix $A = [a_1, a_2, \ldots, a_{n-h+1}]^T$.

Through the gating of attention weight matrix A, the output feature maps are calculated as follows:

$$m_l = C \otimes f(A^l + b)(l = 1, 2, \ldots, t) \tag{9}$$

where $m_l, b \in R^{(n-h+1)\times 1}$, b is a bias matrix, t is the number of convolution kernels we use in the attention-gated layer. We use kernels with different window size k to extract different gained attention weight matrices $A^l (l = 1, 2, \ldots, t)$. Finally, the output feature map is given as follows.

$$M = [m_1, m_2, \ldots, m_t]^T \tag{10}$$

where $b \in R^{(n-h+1)\times 1}$.

The basic operations of MCapsNet-based pest identification method are as follows:

(1) Matrix multiplication of input vector and weight matrix is used to express spatial relations between low-level features and high-level features in the image.
(2) Weighted input vector, where the weight determines which higher capsule the current capsule sends its output to, is accomplished by the dynamic routing process.
(3) Sum the weighted vector, which is the same as the general CNN block.
(4) Use the squash function to compress the vector so that its maximum length is 1 and its minimum length is 0, while keeping its direction unchanged.

4. Experiments and results

In this section, eight common crop pests, such as mucilworms, corn bores, moths, caterpillars, ladybugs, aphids, cotton bollworms and flying cicadas, were studied. The images were collected from the experimental base in Baoji City, Shaanxi Province. In different periods in a natural field environment, nearly 2000 images of pests were collected using image acquisition devices such as smartphones, cameras and the Internet of Things. About 250 images of each pest were collected. At the same time, some network images are used to supplement the data set to ensure its integrity. To improve the training efficiency of the subsequent network model, Photoshop was used to cut the images into JPG colour images with a size of 256×256 pixels. The original pest image examples are shown in Figure 6.

The learning rate plays a decisive role in the convergence of the optimization algorithm of the network model. If the learning rate is too small, the convergence speed is slow. On the contrary, if the learning rate is too large, the model may not converge to the optimal solution. Generally, at the beginning of iterative training, the learning rate is large. As the model gradually converges, the learning rate becomes smaller so that the model can converge to a minimum better to avoid the situation of loss increase. In the process of MCapsNet iteration, the

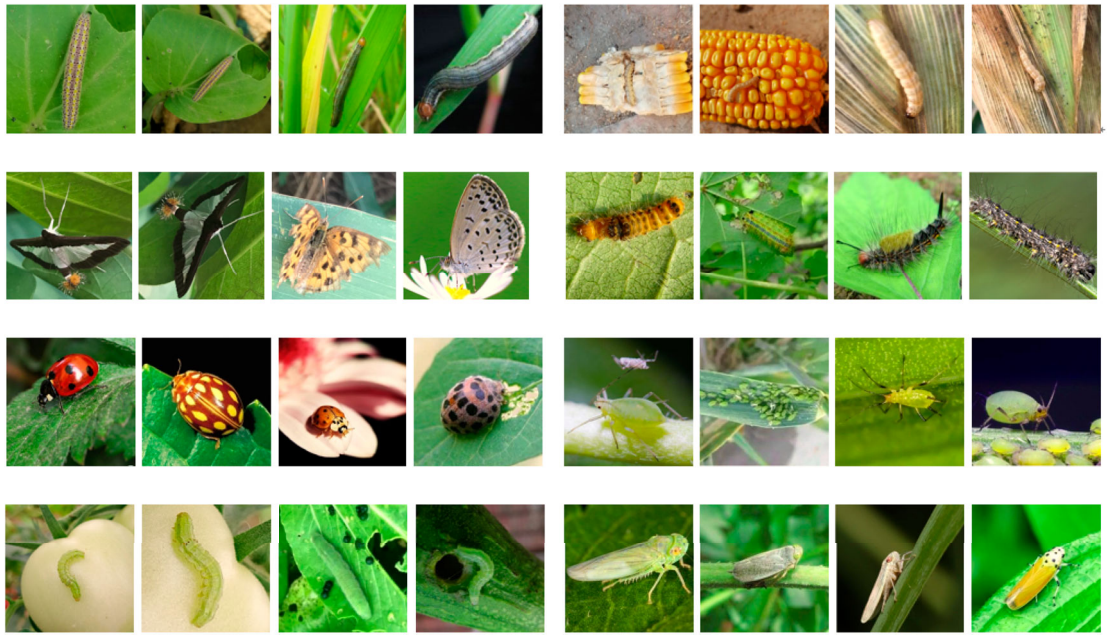


Figure 6. Original crop pest images.

Adam algorithm was used for optimization. A relatively large learning rate was initially selected and set to 0.9, attenuated according to an exponential function with the completion times of data set training.

Crop pest recognition based on MCapsNet mainly includes three parts: dataset preprocessing, MCapsNet training and pest image classification using trained MCapsNet. The main process is described as follows.

(1) Augment each pest image.
(2) Normalize each image after augmentation, and then divide all images into a training set and a test set.
(3) Train MCapsNet with the training set, calculate the weight update by Adam during the iteration and determine whether the weight update is less than the threshold. If it is, the iteration is terminated. Otherwise, keep training. The default threshold is 0.001.
(4) Test the average recognition rate of MCapsNet on crop pest leaf images using the test dataset.

The precision rate (P) and recall rate (R) are usually used to evaluate the performance of the models

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN} \tag{11}$$

where TP is the sentences with correct classification of ADRs, FN is the sentences with ADRs predicted as no ADRs, FP is the sentences without ADRs predicted as ADRs, and TN is the sentences without correct prediction of ADRs.

A lot of experiments are conducted to evaluate the classification performance of the proposed method, where TensorFlow is the deep learning framework, Python3.7 programming development language, and the operating environment of the system is Windows10 64Bit, the hardware development environment is Intel Xeon E5-2643v3 @3.40 GHz CPU, 64GB memory, graphics card NVIDIA Quadro M4000 GPU.

To verify the effectiveness of the proposed algorithm for crop pest detection in the field, the experimental

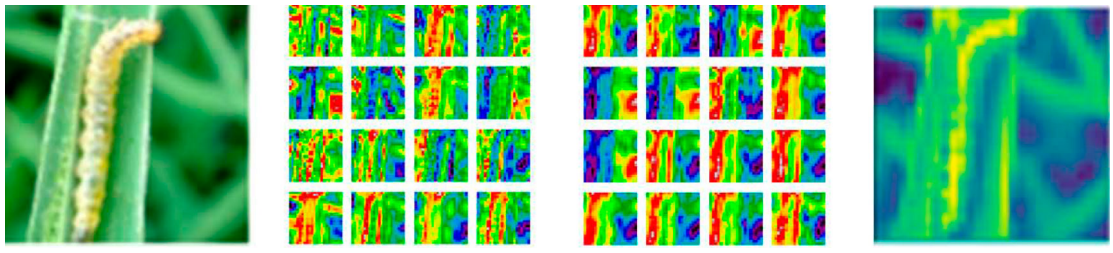


Figure 7. The feature graphs by MCapsNet.

Table 2. The recognition results of crop pests by five methods.

Method	Activation	Precision rate %	Recall rate %
ICNN	ReLU	85.28	76.85
VGG16	ReLU	82.49	72.62
ResNet	ReLU	88.24	80.63
CapsNet	ReLU	75.41	68.30
MCapsNet	ReLU	87.52	78.30
MCapsNet	LeakyReLU	89.52	81.22

results are compared with the other four CNN models (ICNN, VGG16, ResNet and CapsNet) based on the augmented pest image database. LeakyReLU is used as the activation function to ensure the nonlinear ability of the model. The initial learning rate is 0.01, the momentum factor is 0.9, and the Batch size is 128. After 1200 iterations of the model, the learning rate is 0.001. Figure 7 shows the feature graphs by MCapsNet. Table 2 gives the recognition results of crop pests by five methods.

From Table 2, it is found that MCapsNet achieves the highest recognition accuracy and recall rate in the same condition. The main reason is that LeakyReLU and attention mechanism are introduced to improve the performance of MCapsNet, reduce the influence of noise on the model and increase the accuracy of model recognition.

5. Conclusions

Insect pest recognition is vital for food security, a stable agricultural economy and quality of life. To realize rapid crop insect pest recognition, a MCapsNet-based pest identification method is proposed in this paper. The experiments of crop pest image recognition under a complex background are carried on and compared with ICNN, VGG16, ResNet and CapsNet. The results show that the proposed method can better accomplish the crop pest recognition task in nature fields and complex backgrounds, and has higher recognition precision and recall rates. The results show that the proposed method can meet the requirements of crop pest recognition in fields. As a result, the proposed model can be applied in real-world applications and further motivate research on crop disease identification.

Disclosure statement

The authors declare they have no financial or conflict of interest exists in this manuscript.

Funding

This work is supported by the Key science and technology project of the Science and Technology Department of Henan Province (Nos. 212102210406, 212102210404, 222102110134, 222102110280, 222102210122), Science and technology project of the Education Department of Henan Province (Nos. 22B520049, 2021YB0499).

References

Alfarisy A.A., Chen, Q., & Guo, M. (2018). Deep learning based classification for paddy pests & diseases recognition. *International Conference on Mathematics and Artificial Intelligence*, *April* , 21–25. https://doi.org/10.1145/3208788.3208795

Chen, J., Chen, W., Zeb, A., & Nanehkaran, Y. A. (2021). Crop pest recognition using attention-embedded lightweight network under field conditions. *Applied Entomology and Zoology*, *56*(4), 427–442. https://doi.org/10.1007/s13355-021-00732-y

Chen, W. S., Yuen, P. C., Huang, J., & Dai, D. Q. (2005). Kernel machine-based one-parameter regularized fisher discriminant method for face recognition. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, *35*(4), 659–669. https://doi.org/10.1109/TSMCB.2005.844596

Cheng, X., Zhang, Y., Chen, Y., Wu, Y., & Yue, Y. (2017). Pest identification via deep residual learning in complex background. *Computers and Electronics in Agriculture*, *141*(September), 351–356. https://doi.org/10.1016/j.compag.2017.08.005

Deng, L., Wang, Y., Han, Z., & Yu, R. (2018). Research on insect pest image detection and recognition based on bio-inspired methods. *Biosystems Engineering*, *169*(January), 139–148. https://doi.org/10.1016/j.biosystemseng.2018.02.008

Deng, L., Wang, Z., Wang, C., He, Y., Huang, T., Dong, Y., Zhang, X., & Kim, Y. H. (2020). Application of agricultural insect pest detection and control map based on image processing analysis. *Journal of Intelligent & Fuzzy Systems*, *38*(1), 379–389. https://doi.org/10.3233/JIFS-179413

Du, J.-X., Huang, D. S., Wang, X.-F., & Gu, X. (2007). Shape recognition based on neural networks trained by differential evolution algorithm. *Neurocomputing*, *70*(4-6), 896–903. https://doi.org/10.1016/j.neucom.2006.10.026

Du, J.-X., Huang, D. S., Zhang, G.-J., & Wang, Z.-F. (2006a). A novel full structure optimization algorithm for radial basis probabilistic neural networks. *Neurocomputing*, *70*(1–3), 592–596. https://doi.org/10.1016/j.neucom.2006.05.003

Du, X., Huang, D. S., Wang, X.-F., & Gu, X. (2006b). Computer-aided plant species identification (CAPSI) based on leaf shape matching technique. *Transactions of the Institute of Measurement and Control*, *28*(3), 275–285. https://doi.org/10.1191/0142331206tim176oa

Han, F., & Huang, DS. (2006). Improved extreme learning machine for function approximation by encoding a priori information. *Neurocomputing*, *69*(16/18), 2369–2373. https://doi.org/10.1016/j.neucom.2006.02.013

Han, F., Ling, Q.-H., & Huang, D. S. (2008). Modified constrained learning algorithms incorporating additional functional constraints into neural networks. *Information Sciences*, *178*(3), 907–919. https://doi.org/10.1016/j.ins.2007.09.008

Han, F., Ling, Q.-H., & Huang, D. S. (2010). An improved approximation approach incorporating particle swarm optimization and a priori information into neural networks. *Neural Computing and Applications*, *19*(2), 255–261. https://doi.org/10.1007/s00521-009-0274-y

Huang, D. S. (1999). Radial basis probabilistic neural networks: Model and application. *International Journal of Pattern Recognition and Artificial Intelligence*, *13*(7), 1083–1101. https://doi.org/10.1142/S0218001499000604

Huang, D. S., & Du, J.-X. (2008). A constructive hybrid structure optimization methodology for radial basis probabilistic

neural networks. *IEEE Transactions on Neural Networks*, *19*(12), 2099–2115. https://doi.org/10.1109/TNN.2008.2004370

Huang, D. S., & Jiang, W. (2012). A general CPL-AdS methodology for fixing dynamic parameters in dual environments. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *42*(5), 1489–1500. https://doi.org/10.1109/TSMCB.2012.2192475

Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, *147*(1), 70–90. https://doi.org/10.1016/j.compag.2018.02.016

Kromm, C., & Rohr, K. (2020). Inception capsule network for retinal blood vessel segmentation and centerline extraction. *IIEEE 17th International symposium on biomedical imaging*, 1223–1226.

Kumar, V. V., Cherickal, P. J., Nithin, R., & Thomas, T. (2020). An overview of a new neural deep learning network- capsule network. *Journal of Science Technology and Management*, *13*(2).

Li H. C., Ye, W., Pan, L., Li, W., Du, Q., & Tao, R. (2020a). Robust capsule network based on maximum correntropy criterion for hyperspectral image classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *13*(February), 738–751. https://doi.org/10.1109/JSTARS.2020.2968930

Li, Y., Wang, H., Dang L. M., Sadeghi-Niaraki, A., & Moon, H. (2020b). Crop pest recognition in natural scenes using convolutional neural networks. *Computers and Electronics in Agriculture*, *169*(February), Article 105174. https://doi.org/10.1016/j.compag.2019.105174

Liu, J., & Wang, X. (2020). Tomato diseases and pests detection based on improved Yolo V3 convolutional neural network. *Frontiers in Plant Science*, *11*(898). https://doi.org/10.3389/fpls.2020.00898

Miranda, L., Gerardo, B. D., & Tanguilig, B. T. (2014). Pest detection and extraction using image processing techniques. *International Journal of Computer and Communication Engineering*, *3*(3), 189–192. https://doi.org/10.7763/IJCCE.2014.V3.317

Nanni, L., Maguolo, G., & Pancino, F. (2020). Insect pest image detection and recognition based on bio-inspired methods. *Ecological Informatics*, *57*, Article 101089. https://doi.org/10.1016/j.ecoinf.2020.101089

Sabour, S., Frosst, N., & Hinton G, E. (2017). Dynamic routing between capsules. *Advances in Neural Information Processing Systems*, 3856–3866. https://doi.org/10.1016/j.knosys.2012.03.014

Shah, R., Patil, S., Malhotra, A., & Asati, R. (2020). A survey to study about different convolutional neural network on various image classifications. *A Journal of Physical Sciences Engineering and Technology*, *12*(January). https://doi.org/10.18090/samriddhi.v12iSpecial%20Is.18592

Shang, L., Huang, D. S., Du, J.-X., & Zheng, C.-H. (2006). Palmprint recognition using FastICA algorithm and radial basis probabilistic neural network. *Neurocomputing*, *69*(13–15), 1782–1786. https://doi.org/10.1016/j.neucom.2005.11.004

Thenmozhi, K., & Reddy, U. S. (2019). Crop pest classification based on deep convolutional neural network and transfer learning. *Computers and Electronics in Agriculture*, *164*(August), 104906–104906. https://doi.org/10.1016/j.compag.2019.104906

Wang, J., Lin, C., Ji, L., & Liang, A. (2012a). A new automatic identification system of insect images at the order level. *Knowledge-Based Systems*, *33*(3), 102–110. https://doi.org/10.1016/j.knosys.2012.03.014

Wang, J. N., Lin, C. T., Ji, L. Q., & Liang, A.P. (2012b). A new automatic identification system of insect images at the order level . *Knowledge-Based Systems*, *33*(3), 102–110.

Wang, J.-X.-F., & Huang, D. S. (2008). A novel multi-layer level set method for image segmentation. *Journal of Universal Computer Science*, *14*(14), 2428–2452. https://doi.org/10.1145/1346330.1346335

Wang, Q. J., Zhang, S. Y., Dong, S. F., Zhang, G.-C., Yang, J., Li, R., & Wang, H.-Q. (2020). Pest24: A large-scale very small object data set of agricultural pests for multi-target detection. *Computers and Electronics in Agriculture*, *175*(1), Article 105585. https://doi.org/10.1016/j.compag.2020.105585

Wang, X.-F., & Huang, D. S. (2009). A novel density-based clustering framework by using level set method. *IEEE Transactions on Knowledge and Data Engineering*, *21*(11), 1515–1531. https://doi.org/10.1109/TKDE.2009.21

Wang, X.-F., Huang, D. S., & Xu, H. (2010a). An efficient local Chan-Vese model for image segmentation. *Pattern Recognition*, *43*(3), 603–618. https://doi.org/10.1016/j.patcog.2009.08.002

Wu, J., Li, B., & Wu, Z. (2019a). Detection of crop pests and diseases based on deep convolutional neural network and improved algorithm. *4th International conference on machine learning technologies*, 20–27. https://doi.org/10.1145/3340997.3341010

Wu, X., Zhan, C., Lai, Y. K., Cheng, M. M., & Yang, J. (2019b). IP102: A large-scale benchmark dataset for insect pest recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, 8787–8796. https://doi.org/10.1109/CVPR.2019.00899

Xiao-Feng, W., Huang, D.-S., Du, J.-X., Xu, H., & Heutte, L. (2008a). Classification of plant leaf images with complicated background. *Applied Mathematics and Computation*, *205*(2), 916–926. https://doi.org/10.1016/j.amc.2008.05.108

Zhang, H. T., Hu, Y.X., & Zhang, H.Y. (2013). Extraction and classifier design for image recognition of insect pests on field crops. *Advanced Materials Research*, *756-759*, 4063–4067. https://doi.org/10.4028/www.scientific.net/AMR.756-759.4063

Zhang, J., Wang, R., Xie, C., & Li, R (2014). Crop pests image recognition based on multifeatures fusion. *Journal of Computational Information Systems*, *10*, 5121–5129. https://doi.org/10.12733/jcis10592

Zhang, Q., Yang LT., Chen, Z., & Li Peng. (2018). A survey on deep learning for big data. *Information Fusion*, *42*(July), 146–157. https://doi.org/10.1016/j.inffus.2017.10.006

Zhang, Z., Ye, S., Liao, P., Liu, Y., & Sun, Y. (2020). Enhanced capsule network for medical image classification. *42nd Annual International Conference of the IEEE Engineering in medicine & biology society (EMBC)*, Canada: Montreal, 1544–1547.

Zhao, Z.-Q., Huang, D. S., & Sun, B.-Y. (2004). Human face recognition based on multiple features using neural networks committee. *Pattern Recognition Letters*, *25*(12), 1351–1358. https://doi.org/10.1016/j.patrec.2004.05.008