International Journal of Pattern Recognition
and Artificial Intelligence
Vol. 34, No. 13 (2020) 2052010 (20 pages)

DOI: 10.1142/S0218001420520102

World Scientific
www.worldscientific.com

Classification of Ecological Data by Deep Learning

Shaobo Liu*, Frank Y. Shih*,¶, Gareth Russell†,
Kimberly Russell‡ and Hai Phan§

*Department of Computer Science
New Jersey Institute of Technology
Newark, NJ 07102, USA

†Department of Biological Sciences
New Jersey Institute of Technology
Newark, NJ 07102, USA

‡Department of Ecology and Natural Resources
Rutgers University, New Brunswick, NJ 08901, USA

§Department of Information Systems
New Jersey Institute of Technology
Newark, NJ 07102, USA
¶shih@njit.edu

Received 10 August 2019
Accepted 12 December 2019
Published 4 May 2020

Ecologists have been studying different computational models in the classification of ecological species. In this paper, we intend to take advantages of variant deep-learning models, including LeNet, AlexNet, VGG models, residual neural network, and inception models, to classify ecological datasets, such as bee wing and butterfly. Since the datasets contain relatively small data samples and unbalanced samples in each class, we apply data augmentation and transfer learning techniques. Furthermore, newly designed inception residual and inception modules are developed to enhance feature extraction and increase classification rates. As comparing against currently available deep-learning models, experimental results show that the proposed inception residual block can avoid the vanishing gradient problem and achieve a high accuracy rate of 92%.

Keywords: Deep learning; convolutional neural network; image augmentation; ecology; bee wings; image classification; inception residual module.

1. Introduction

Deep-learning model has been widely used in image processing, computer vision, and pattern recognition. It is a branch of machine learning based on applying a large amount of data for training a model. Different from traditional machine learning methods, the deep-learning model can learn features automatically from vast amount

¶Corresponding author.

of data samples and does not require domain expert's assistance in building feature extractor. It can be categorized into supervised and unsupervised learning, to be applied for various tasks on respective types of data. For instance, one can apply the Convolutional Neural Network (CNN) for image classification or the Recursive Neural Network (RNN) for language processing. In computer vision, CNN is an effective framework to recognize and classify different targets.

The CNN model was initially proposed by LeCun[15] as LeNet in 1995. Due to the limited computing power and incomplete mathematical proof, LeNet was difficult to be accepted by most of researchers. While with recent improvements in computing capacity and speed, CNN model has achieved better performance than traditional machine learning methods in various fields such as object classification, object detection, natural language processing, etc.

AlexNet[14] was developed in 2012 with a more complex structure than LeNet. AlexNet contains millions of parameters. It is constructed by five convolution layers, including max-pooling layer, dropout layer and three fully-connected layers. AlexNet won the championship in the 2012 ImageNet competition with the test error rate of 15.4%. In 2014, the Visual Geometry Group at University of Oxford proposed the VGG network,[24] which constructs the network with more convolutional layers. Microsoft Research Asian proposed the ResNet,[10] which achieved the test error rate of 3.6% in the 2015 ImageNet competition. It is used a residual block to avoid the vanishing gradient problem in back propagation. However, it took two to three weeks to finish training on an 8-GPU machine.

The Google company proposed GoogLeNet[25] with 22 layers. GoogLeNet achieved a promising result with error rate of 6.7% on the 2015 ImageNet competition. Its inception module uses different kernel sizes in feature extraction and reconstructs the feature maps using 1×1 convolution.[16] Later, more advanced versions have been developed, including inception v2,[11] inception v3,[26] and inception v4.[27] The inception v2 introduces batch normalization to produce a fixed distribution for the output of each layer. The inception v3 and inception v4 adopt factorizing convolution to reduce the parameters of convolutional blocks. The inception residual block[26] was constructed by adding a residual connection between activation functions for solving the vanishing gradient problem.

In ecology, one task is to identify bee species from bee-wing images. Traditional machine learning methods, such as random forest, artificial neural networks, support vector machines, and genetic algorithms,[2–5,7,9,23,28] have already been applied on classification of ecological image data. Researchers adopted support vector machines, artificial neural networks, Naïve Bayes,[12] k-nearest neighbors, [22] and logistic classifier[31] in classifying bee wings. However, the combination of deep learning and ecology has seldom been explored. Ecologists have shown increasing interests in building more efficient species classification systems using deep-learning neural networks. In other application of deep learning and ecology, Schneider *et al.*[19] used the RNN to classify different types of animals from trap camera data.

Species are various in nature and one specie usually has different kinds of subspecies. Two problems occurred in training CNN models on the ecological data are the limited image dataset and imbalanced sample numbers among different classes. Therefore, the test accuracies for the existing CNN models on the ecological dataset are low. In this paper, we utilize three methods to increase the test accuracy in classifying the ecological dataset. The first method uses data augmentation[6,8,17,33] to enlarge the dataset by performing transformation operations. The second method adopts transfer learning[1,13,20,21,30] by applying the parameters of a pretrained CNN model to the proposed classification task. Transfer learning uses a highly generalized knowledge and transfers this knowledge to learn the ecological datasets. As compared with model training in random initialization, the pretrained model converges much faster and achieves better performance. The third method combines data augmentation with transfer learning, so the test accuracy in ecological datasets can be further improved.

The rest of this paper is organized as follows. Section 2 introduces the deep-learning framework, including LeNet-5, AlexNet, VGG, residual neural network, and inception models. Section 3 presents the classification in the ecological dataset. Section 4 describes the improvement on the test accuracy. Section 5 shows experimental results. Finally, conclusions are drawn in Sec. 6.

2. Deep Learning Framework

Deep learning requires a large amount data to train and evaluate learning model's performance. Figure 1 shows the structure of LeNet-5, which is initially intended for the classification of hand-written digits. It is composed by several layers with different functions. Like other machine learning models, LeNet-5 needs a feature representation method to compress one (i.e. a grayscale image) or three (i.e. an RGB image) 2D matrices into feature representation.

In LeCun's design, LeNet-5 contains an input layer, a convolution layer to extract features, and a pooling layer to reduce unnecessary data. After a second connection of a convolutional layer and a pooling layer, feature representation is fed to a fully

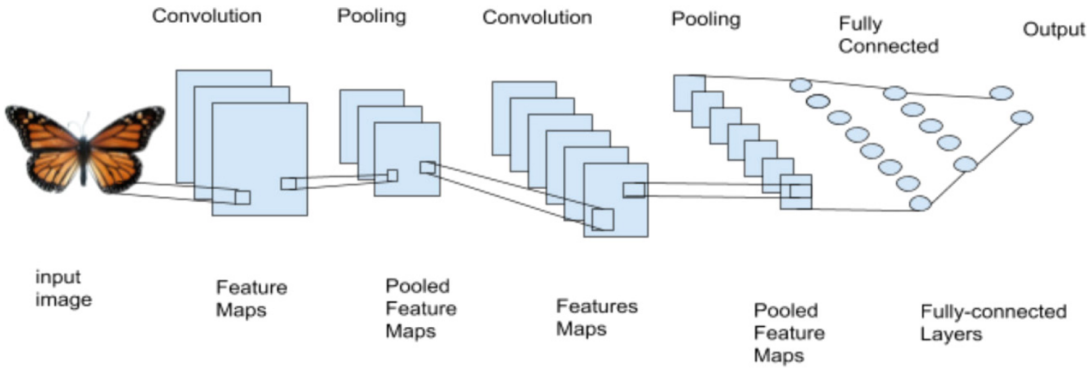


Fig. 1. The structure of LeNet-5.

connected artificial neural networks for classification. In the convolutional layer, the input is one or several images with one or three channels. We perform convolution several times with different filters, so there are several output images, called feature maps. The convolutional layers extract different local features using different filters to learn all the features. The convolutional layer followed by an activate function is described as

$$h^k = f\left(\sum_{l \epsilon L} x^l \otimes w^k + b^k\right), \tag{1}$$

where h^k is the latent representation of kth feature map of the current layer, f is the activation function, x^l is the lth feature map of group of feature maps L of the previous layers or the lth channel of the input images with totally L channels in the case of the first layer of the network, $\otimes$ denotes the 2D convolution operation, and w^k and b^k denote the weights (filters) and biases of the kth feature map of the current layer, respectively. A nonlinear function, called ReLU (Rectified Linear Unit), works as the activation function f, which can be written as $f(x) = \max(0, \mathrm{x})$. This function will stay 0 when x is less than 0 but return to be x for any positive input. The ReLU works well for neural network models because it allows the models to compute nonlinearities and interaction.

Let a SoftMax function be defined as

$$p_i = \frac{r^{z_i}}{\sum_{k=1}^{K} e^{z_k}}, i = 1, 2, 3, \ldots, K, \tag{2}$$

where z_i is an element of the input tensor. Using the SoftMax function, we can transform an N-dimensional vector of real numbers into a vector of real numbers in the range of (0, 1). The loss function is the cross-entropy, which is denoted as

$$H(y, p) = -\sum_i y_i \log(p_i), \tag{3}$$

where y_i is the label of ith input image and p_i is the ith element of the output of SoftMax function.

The pooling layer is designed to perform down-sampling on image data to reduce the size of feature maps. There are two typical pooling methods: average pooling and max-pooling. Average pooling is used to compute the average value in a small area and max-pooling is used to extract the maximum value. After sufficient information is acquired from convolutional layers and pooling layers, the fully connected layer is used to map the output to linearly separable space and flatten the matrix into a vector. Then, SoftMax is used for regression to classify the data, so the output of the last fully connected layer is the predicted label.

AlexNet[14] is the first deep CNN with five convolutional layers using ReLu as an activation function. In feature extraction, AlexNet uses large convolution kernels to

extract features. In ILSVRC 2010, AlexNet obtained the top-1 and top-5 error rates of 37.5% and 17.0%, respectively.

VGG neural network[24] was developed by Visual Geometry Group, University of Oxford. In the 2014 ILSVRC (ImageNet Large Scale Visual Recognition Competition), VGG-16 obtained an error rate of 8.8% and VGG-19 obtained an error rate of 9.0%. In the VGG model, stacked convolution kernels with 3-by-3 are used. Note that two 3-by-3 convolution kernels equal to a 5-by-5 effective convolution area, three 3-by-3 kernels equal to a 7-by-7 effective area, and so on. The purpose of using stack convolutions is to reduce parameters in the learning process. The VGG16 contains two 5-by-5 convolutional layers and three 7-by-7 convolutional layers and the VGG19 contains two 5-by-5 convolutional layers and three 9-by-9 convolutional layers. However, when more convolution layers are stacked together, a vanishing gradient problem may happen. It is occurred during backpropagation when several small derivatives are multiplied together after the same activation function. The problem of a small gradient will cause the parameters not to be updated effectively.

In order to solve the vanishing gradient problem, a new convolutional block, called residual block, is introduced in residual neural network.[10] By adding a shortcut connection between the input x to learn residual mapping $F(x)$ before the activation function, the output $x + F(x)$ is able to maintain a higher overall derivative. With residual connections, the residual neural network can add up to 152 layers. It won the competition in 2015 ILSVRC.

The inception block was introduced by GoogleNet,[25] which uses different kernel sizes. In inception block, 1×1 convolution, 3×3 convolution, 5×5 convolution, and 3×3 Max-pooling are used at the same time using the same convolution. The 1×1 convolution with ReLu activation works as dimension reduction to reconstruct the feature maps.[16] Figure 2 shows the inception block in GoogleNet.[25]

GoogleNet (Inception v1) is the first version of inception model, continued by Inception v2,[11] Inception v3,[26] and Inception v4.[27] Inception v1 is the winner of 2014 ILSVRC (ImageNet Large Scale Visual Recognition Competition). Inception v2

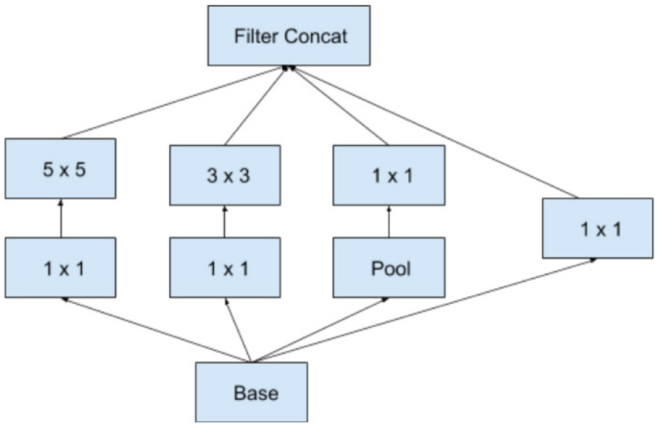


Fig. 2. Inception module with dimension reduction.

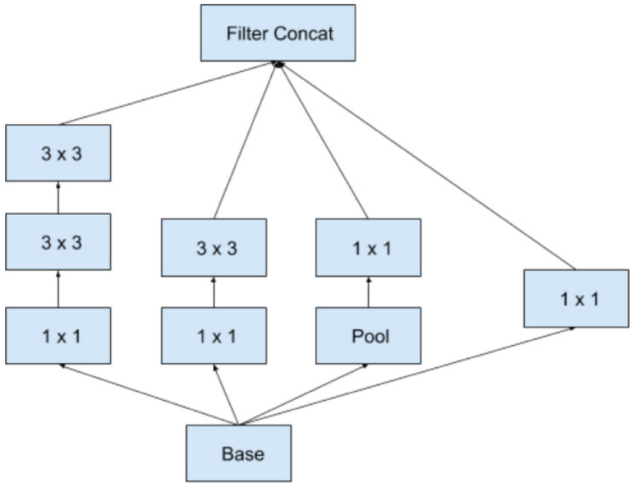


Fig. 3. Factorization into a smaller convolution.

introduced a concept termed as batch normalization to normalize the value distributions of layers' output and keep the distribution fixed. Inception v3 introduced factorizing convolution to reduce parameters. Two kinds of factorizing convolutions were introduced in Ref. 26, where small kernel convolutions are used to replace large convolutions and asymmetric convolutions are used to replace symmetric convolutions. Figure 3 shows a factorization in smaller convolutions, where the 5×5 convolution area is replaced by two 3×3 convolution areas.

Asymmetric convolution with one $n \times 1$ followed by one $1 \times n$ convolution can replace a $n \times n$ area. The purpose of using asymmetric convolution is to reduce the number of operations and maintain the network's efficiency. With asymmetric convolution, a new version of inception module is shown in Fig. 4.

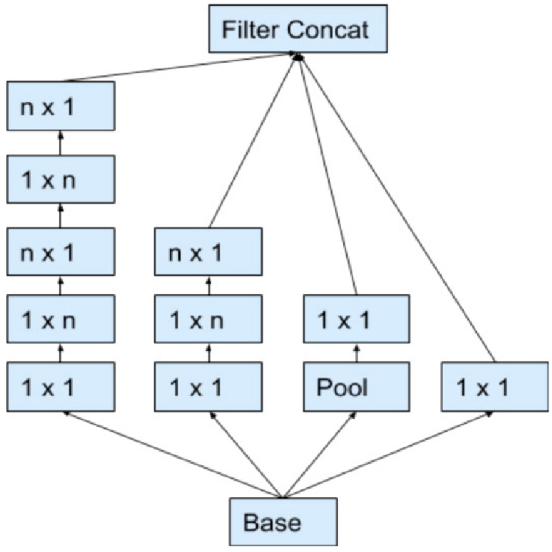


Fig. 4. Factorization into asymmetric convolution.

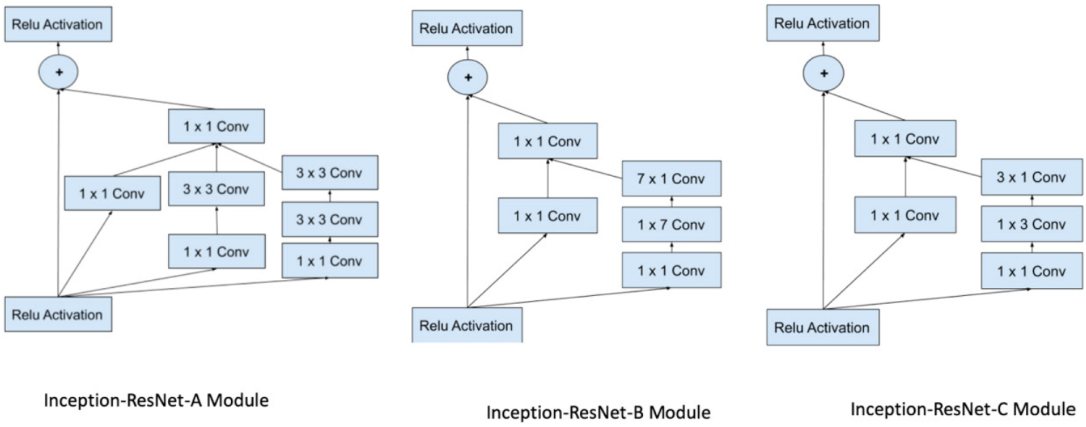


Fig. 5. Inception-residual modules in inception-residual v2.

The Inception-ResNet-v1 and Inception-Resnet-v2 were introduced in Inception v4,[27] where a shortcut connection is added between two activation functions. Three Inception residual blocks in Inception-ResNet-v1 and Inception Resnet-v2 are shown in Fig. 5.

3. Classification in Ecological Dataset

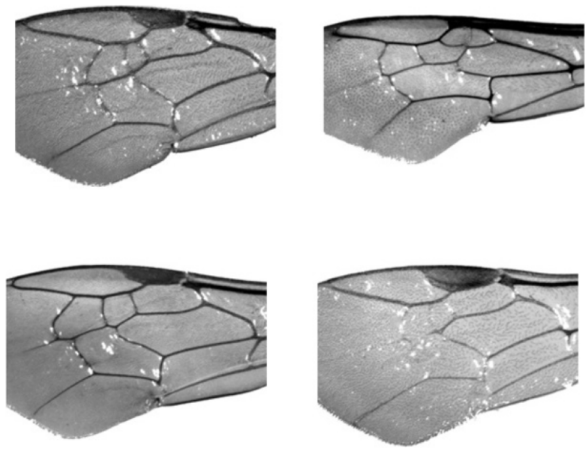
3.1. *Ecological Dataset*

Two ecological datasets of bee-wing and butterfly are used. The bee wing dataset is a relatively small and unbalanced dataset. There are 19 types of bee wing, with totally 755 samples, including 566 images for training and 189 images for testing. Within the 19 classes, 8 main classes are agapostemon, augochlora, augochlorella, augochlorella, ceratina, dialictus, halictus, and osmia. Ceratina contains three subclasses, which are *Ceratina calcarata*, *Ceratina dupla*, and *Ceratina metallica*. Dialictus contains four subclasses, which are Dialictus bruneri, Dialictus illinoensis, *Dialictus imitatus*, and *Dialictus rohweri*. Halictus contains two subclasses, which are *Halictus confusus* and *Halictus ligatus*. Osmia contains five subclasses, which are *Osmia atriventis*, *Osmia bucephala*, *Osmia cornifrons*, *Osmia georgica*, and *Osmia lignaria*. The samples among subclasses are similar but difficult to be distinguished by human beings. In Fig. 6, (a) shows some samples in bee wing dataset and (b) shows the sample distribution in all classes.

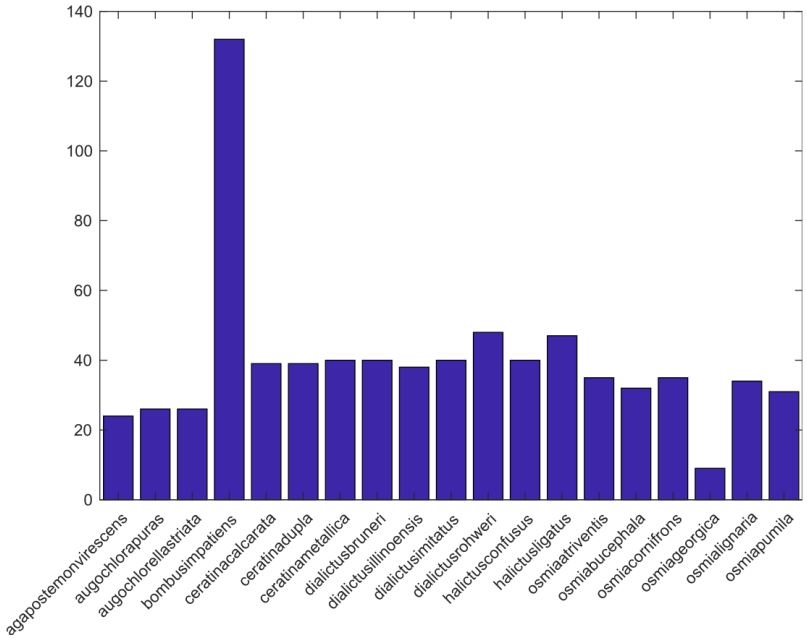
The butterfly dataset[29] is a small and relatively balanced dataset. It contains 10 classes of RGB butterfly species, ranging from 55 to 100 images per class. The total dataset contains 832 images, where 627 images are used for training and 205 images for testing. In Fig. 7, (a) shows some images in the butterfly dataset and (b) shows the sample distribution in all classes.

3.2. *Classification Within Original Dataset*

Deep-learning models are mainly designed for large datasets such as ImageNet. Considering the ecological datasets to be relatively small, we test and evaluate the



(a)



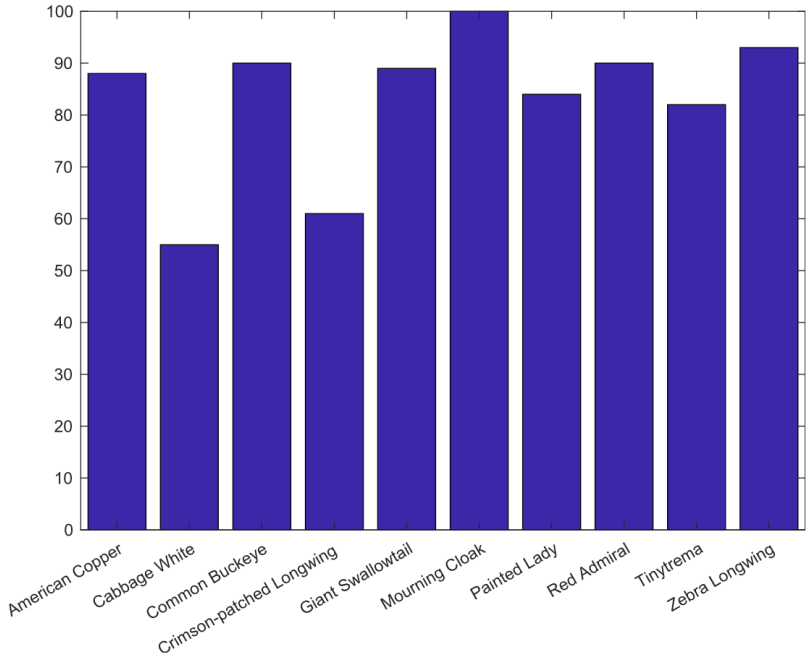
(b)

Fig. 6. (a) Samples in bee wing dataset, (b) sample distribution in all classes.

ecological datasets using seven convolutional models, including LeNet-5, AlexNet, VGG-16, VGG-19, Residual Net 50, InceptionV3, and Inception Residue V2. The test results are listed in Table 1.

A similar test accuracy of 87% is obtained by LeNet-5, AlexNet, ResNet50, Inception v3, and InceptionResNetV2, except by VGG-16 and VGG-19. LeNet-5 is a two-layer CNN, which can extract features in the bee wing dataset. AlexNet uses five convolutional layers but obtains a lower test accuracy of 86%, indicating that a vanishing gradient problem may occur.

(a)



(b)

Fig. 7. (a) Samples in butterfly dataset, (b) sample distribution in all classes.

Table 1. Test accuracy on the ecological datasets.

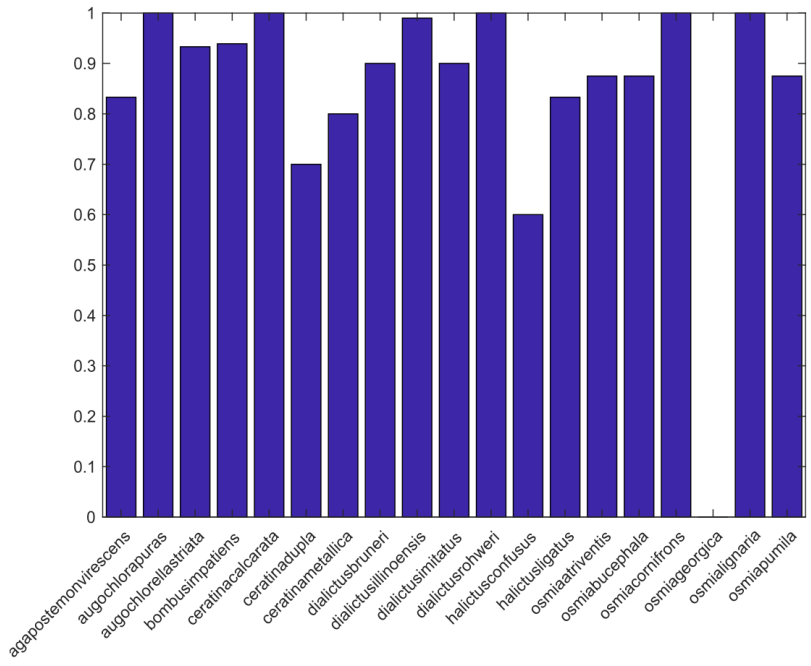
	Bee Wing (%)	Butterfly (%)
LeNet-5	87.78	70.24
AlexNet	86.04	79.85
VGG16	17.74	12.17
VGG19	17.72	12.28
ResNet50	86.54	75.36
Inception v3	87.16	78.84
InceptionResNetV2	87.72	79.98

The VGG-16 and VGG-19 models suffer a convergence problem in training which could be caused by a small size of data samples. The ResNet50 consists of more stacked convolutional layers than the VGG models. It uses a residual connection to avoid the vanishing gradient problem. Inception v3 uses an inception block with

different convolution kernel sizes to enrich the feature map. Inception residual neural network combines inception blocks with residual connection.

The test accuracy on the bee wing dataset is analyzed as follows. The test accuracy of each class is shown in Fig. 8(a). From Fig. 8(b), a relatively lower test accuracy is observed between subclass species. In Ceratina, the classification of *Ceratina dupla* subclass obtains a test accuracy of 70%, which is 17% lower than the overall accuracy. In Halictus, the classification of *Halictus confusus* subclass has a test accuracy of 60%, which is 27% lower than the overall accuracy. In Osmia, the classification of *Osmia georgica* subclass has a test accuracy of zero, since the two samples are classified as the *Osmia georgica*, which is another subclass of Osmia. Figure 8(c) shows a heap map of the confusion matrix.

Using AlexNet and InceptionResV2 models, two similar test accuracies of 79% are obtained on the butterfly dataset. Using LeNet, a lower test accuracy of 70% is obtained due to its insufficient number of convolution layers. The VGG16 and VGG19 models face a similar convergence problem in the bee wing dataset. Using ResNet50, a test accuracy of 75% is obtained, indicating that the residual connection makes the model to go deeper. Note that the InceptionResV2 model obtains a higher test accuracy than InceptionV3, showing a promising feature extraction ability for the inception residual block.



(a)

Fig. 8. (a) Test accuracy in the bee wing dataset, (b) bee-wing subclass classification, (c) heatmap of confusion matrix. Note that labels are from 1 to 19, respectively, representing from *Agapostemon virescens* to *Osmia pumila*.