



Short Communication

# Shallow convolutional neural network for image classification

Fangyuan Lei<sup>1,2</sup>  · Xun Liu<sup>1</sup> · Qingyun Dai<sup>1,2</sup> · Bingo Wing-Kuen Ling<sup>3</sup>

Received: 4 August 2019 / Accepted: 12 December 2019 / Published online: 17 December 2019  
© Springer Nature Switzerland AG 2019

## Abstract

Deep convolutional neural networks show great advantages in computer vision tasks, such as image classification and object detection. However, the networks have complex network structure which include a large number of layers such as convolutional layers and pooling layers. They greatly consume valuable computing and memory resources, and also hugely waste training time. Therefore, we propose a novel shallow convolutional neural network (SCNNB) to overcome the above limitations for image classification, which uses batch normalization techniques to accelerate training convergence and improve the accuracy. The SCNNB network has only 4 layers with small size of convolution kernels, which requires low time complexity and space complexity. In the experiments, we compare the SCNNB model with two variant models and the classical SCNN model on the two benchmark image datasets. Experimental results show that compared to SCNN model, the SCNNB model can quickly learn the features of the data and achieve the highest classification accuracy of 93.69% with 3.8 M time complexity on fashion-MNIST.

**Keywords** Deep convolutional neural networks · Shallow convolutional neural network · Batch normalization · Image classification

## 1 Introduction

Letnet-5 network is a highly efficient convolutional neural network, which has a network structure with 7 layers including pooling layers, has been successfully used to handwritten character recognition [1]. When the problem of gradient disappearance in deep network training is solved by Hinton [2], the Deep CNN algorithm is rapidly developed for different applications. With abundant computing resources and deep layers, the DCNNs obtain better performance. From 8 layers AlexNet [3] to 152 layers ResNet152 [4], there is tendency that the layers of DCNN structure become more and deeper. However, training those deep networks models occupies massive computation and memory resources. Especially, the training procedure of the DCNN model is very time consuming.

Compared with DCNN, shallow convolutional neural network (SCNN) has a simpler network structure and fewer network parameters. Therefore, training the SCNN model occupies fewer computation resources and memory. To achieve better performance, Agarap [5] proposes a shallow network that combines CNN with support vector machine (SVM). Lee et al. [6] propose a shallow CNN with logarithmic filter groups to reduce model size for classification tasks. In order to reduce the computational cost, a fast-shallow CNN is proposed [7, 8]. However, in the traditional deep neural network training, there is an internal covariate shift phenomenon [4]. It means that during the update of the training parameters, the data distribution of each input in the middle layer tends to be quite different from that before the parameter update. This phenomenon causes the network to constantly adapt to the new data

✉ Fangyuan Lei, Leify@126.com; Xun Liu, 1834720998@qq.com; Qingyun Dai, 1144295091@qq.com; Bingo Wing-Kuen Ling, yongquanling@gdut.edu.cn | <sup>1</sup>School of Electronic and Information, Guangdong Polytechnic Normal University, Guangzhou 510640, China. <sup>2</sup>Guangdong Key Provincial Laboratory of Intellectual Property and Big Data, Guangdong Polytechnic Normal University, Guangzhou 510665, China. <sup>3</sup>School of Information Engineering, Guangdong University of Technology, Guangzhou 510006, China.



distribution, which makes training extremely difficult. The batch normalization (BN) strategy [12] is introduced to deal with those parameters change during the shallow CNNs training [9–11]. The BN strategy accelerates network convergence and improves the generalization ability of the SCNN model. In order to improve image classification accuracy, the network structure of these shallow CNNs should be further optimized.

In this article, we propose a novel shallow convolutional neural network (SCNNB) with batch normalization (BN) for image classification. The SCNNB network consists of two convolutional layers, two max-pooling layers, one fully connected layer and one softmax layer. To accelerate the convergence of the network and improve the generalization ability of the model, the BN is added after each convolutional layer of the SCNNB network.

The main contributions of this paper are as follows:

1. We propose a novel shallow convolutional neural network (SCNNB) with batch normalization technology to accelerate convergence and improve accuracy.
2. Without pre-training, the SCNNB model could achieve high accuracy than VGG [13] on image classification.
3. The SCNNB model has a simple structure, which contains four layers (pooling layer and BN not considered), and the size of convolution kernels is only  $3 \times 3$ . The network model has low time and spatial complexity.

## 2 Related work

Convolutional neural network (CNN) uses strategies such as weight sharing and pooling to greatly reduce the amount of calculation and parameters, which makes the training depth model possible. In recent years, with the great increase in computing power, many deep convolutional neural networks (DCNNs) are rapidly developed. From AlexNet [3], VGGNet [13] to MobilenetV1 [14] and MobilenetV2 [15], they have been proved that DCNNs have achieved gratifying successes in computer vision tasks. However, these DCNNs have complex network structures and a large number of parameters. For example, VGG16 [13] has 16 layers, and over one million parameters.

Due to the time-consuming training of deep networks, the research work on shallow deep learning networks has received attention [16–20]. In [21], the author proposes a shallow CNN to face detection, which has only a convolutional layer and max-pooling layer. Niu et al. [22] propose an end-to-end shallow CNN, which combines regression and CNN with several layers to process ordinal regression on two age benchmark datasets. These models construct a non-linear mapping from input to output. They use convolution technology to extract data features. They utilize

pooling strategy to reduce parameters, and fuse the features of the front layer with fully connected layer. In order to better extract features and accelerate the training of shallow CNN models, Bhatnagar et al. [9] apply the ideas of BN [12] and residual skip connections [4] to classify the fashion-MNIST datasets. In [5], the author combines support vector machine (SVM) with shallow CNN to image classification. In their network structure model, the fully connected layer is replaced by the SVM. The simulation results show that the classification is better than single CNN and SVM on MNIST and fashion-mnist datasets. Lee et al. [6] propose shallow CNN, which uses several logarithmic filter groups convolutions and global average pooling to get more accuracy in computer visual tasks. In [7], the author proposes a fast shallow CNN without pooling layer to detect forgery image, which extracts CrCb channels from RGB space.

## 3 Methods

In this section, a novel shallow convolutional neural network (SCNNB) framework is proposed, then the feature of the SCNNB model is deeply analyzed. Moreover, the time complexity of the convolutional neural network (CNN) model is evaluated.

### 3.1 SCNNB model

Deep convolutional neural networks (DCNNs) such as MobilenetV1 [14] and MobilenetV2 [15] have a large number of layers and require several days, weeks or even longer training time. To avoid the above problem, we construct a shallow CNN framework with fewer layers and small convolution kernels size. The shallow convolutional neural network (SCNNB) with batch normalization model is composed of 2-layer convolutional layers, 2-layer max-pooling layers, a fully-connected layer and a softmax layer. The model framework is shown in Fig. 1. The size of the input data is  $28 \times 28 \times 1$ . The model first extracts the shallow data features by  $3 \times 3$  convolution with 32 filters. In CNN, the batch normalization (BN) is to normalize (average and variance) each feature map obtained after convolution so that the input value of the activation function falls within the range sensitive to the input to reduce the probability of gradient vanishing. In order to improve accuracy and the nonlinear expression ability of the model, the convolutional layer is followed by BN and Relu. Then  $2 \times 2$  max-pooling is used to reduce data dimension and computational complexity. Next  $3 \times 3$  convolution with 64 filters to extract the deep data features. After the second convolutional layer, there is a pair of strategies BN and Relu which further improve

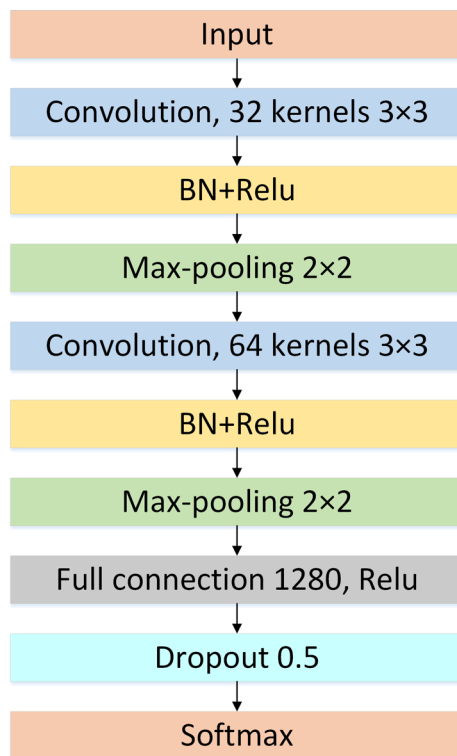


Fig. 1 Architecture structure for our SCNNB model

accuracy and increase the nonlinearity of SCNNB model. The strategies followed by the second  $2 \times 2$  max-pooling layer further reduce data dimension and computational complexity. Then a fully connected layer with 1280 neurons is used to fuse the features of the front layer. After the fully connected layer, Relu is used to increase the nonlinear expression ability of the model. Next dropout is introduced to reduce over-fitting and improve the generalization ability of the model. Finally, softmax output layer to achieve multi-classification. It requires small computations, memory and few iterations, saving valuable time resources.

The SCNNB model has several modules including input layer, convolutional layer, max-pooling layer, fully-connected layer and softmax output layer, as well as BN, dropout strategy and Relu activation function.

1. Input layer: The size of the input images is  $28 \times 28$  for single channel image.
2. Convolutional layer: Convolutional layer is one of the most important layers of CNN, which can effectively extract the characteristics of data. Different convolution kernels extract different data features. The more convolution kernels, the stronger the ability to extract features is. The SCNNB contains two  $3 \times 3$  convolutional layers which have 32 and 64 filters, respectively.

3. Max-pooling layer: After the features of data are extracted by convolution, the pooling layer is used to reduce data redundancy through down-sampling on the extracted features. The SCNNB uses two  $2 \times 2$  max-pooling layers in order to reduce the data dimension and the computational complexity while keeping the useful features of the extraction almost unchanged. The max-pooling layers output a matrix whose output channel is the same size as the input channel. The pooling layers not only reduce the redundancy of data features and the risk of over-fitting, but also improve the training speed.
4. Fully connected layer: Each neuron node of the fully connected layer is connected to each neural node of the upper layer, and the neuron nodes of the same layer are disconnected. The SCNNB model fuses the features of the front layer by a fully connected layer with 1280 neurons. This fully connected layer is essentially  $1 \times 1 \times 3136$  convolution operation, which convolution kernels size is same with the output characteristic size ( $7 \times 7 \times 64$ ) of the previous layer.
5. Softmax output layer: The most commonly used convolutional neural network uses softmax output layer to achieve multi-classification. Softmax function is defined as:

$$\text{softmax}(y)_i = \frac{e^{y_i}}{\sum_{j=1}^n e^{y_j}} \quad (1)$$

where  $n$  indicates the number of output layer nodes, corresponding to the number of categories of the specific classification task, and  $y_i$  denotes the output of the  $i$ th node of the output layer. The output of this model is converted into a probability distribution by the softmax function.

6. Relu: The Relu is chosen as the non-linear activation function of the SCNNB model. Relu could learn from data the mapping of any complex function from input to output to solve the non-linear problems, and make the model more powerful. The Relu activation function is defined as:

$$f(x) = \max(0, x). \quad (2)$$

Relu has the property of sparse activation, which alleviates the over-fitting to some extent and improves the generalization ability of the model.

7. BN: SCNNB uses BN strategy to speed up the training of the model and improve the classification results. The details of BN are introduced in Sect. 3.1.
8. Dropout: In the model of deep learning, if there are too many training parameters and fewer input data, it will probably lead to the problems of high training accuracy and low testing accuracy, such as overfitting.

SCNNB uses dropout technique to randomly discard the neurons of a certain probability on the fully connected layer to avoid over-fitting and accelerate the training of the network.

### 3.2 Time complexity assessment

The time complexity of convolutional neural network includes convolutional layers, pooling layers and fully connected layers. The pooling layers and fully connected layers only take 5–10% of the computational time [23], while the convolutional layers occupy the vast majority of computing time. As with the idea of Lu et al. [10], the SCNNB model only considers the time complexity of the convolutional layers to simplify the calculation. According to [23], we calculate the theoretical time complexity which is defined as:

$$O = \left\{ \sum_{j=1}^k n_{j-1} \cdot s_w \cdot s_h \cdot n_j \cdot m_w \cdot m_h \right\}. \quad (3)$$

where  $j$  denotes the index of the convolutional layer, and  $k$  is the number of the convolutional layers.  $n_{j-1}$  denotes the number of the filters (input channels) in the  $j - 1$ th layer, and  $n_j$  is the number of the filters (output channels) in the  $j$ th layer.  $s_w$  and  $s_h$  are the width and height of the filters, respectively, and  $m_w$  and  $m_h$  are the width and height of the output feature map, respectively.

## 4 Experiments

In this section, firstly two variants of SCNNB model are compared with SCNNB model on MNIST, Fashion-MNIST, and CIFAR10 datasets. Then the SCNNB model is compared with classic deep convolutional neural networks and shallow convolutional neural networks methods.

### 4.1 Datasets

MNIST [1] is a standard dataset for image classification. It has a total of 10 classes, including 10 digits of 0–9. We choose 60,000 grayscale images for training and 10,000 grayscale images for testing. The size of the images is 28

× 28 pixels. Fashion-MNIST datasets [14] is a new image dataset, which consists of a total of 70,000 fashion products images from 10 categories. The same as MNIST, the fashion-MNIST dataset contains 60,000 training images and 10,000 test images. The images are grayscale images of the size of 28 × 28. We randomly flip the training and testing images according to the probability of 0.5, and use them as our training set and testing set. CIFAR10 [27] is widely used image classification dataset, which has 50,000 training color images and 10,000 test color images from 10 categories. The size of these images is 32 × 32 pixels.

### 4.2 Experimental parameters

In the experiments, the model parameters are updated by stochastic gradient descent (SGD) with momentum. We set fixed learning rate of 0.02 and the momentum of 0.9. Dropout rate is 0.5, and regularization weight is 0.000005. All experiments are trained for 300 epochs, and batch size is 128.

### 4.3 Results

In order to prove that BN can accelerate the training of network and improve precision, we introduce two variants of SCNNB as follows:

SCNNB-a: remove only BN strategy after the first convolutional layer, and the remaining layers and parameters remain unchanged.

SCNNB-b: remove all BN strategies after each convolutional layer, and the remaining layers and parameters remain unchanged.

The comparison results between SCNNB, SCNNB-a and SCNNB-b are shown in Table 1. The results in Table 1 show that on MNIST dataset, SCNNB achieves the highest test accuracy of 99.54%, which is 0.06% higher than SCNNB-a. The SCNNB is 0.08% higher than SCNNB-b.

The results in Table 1 depict that on fashion-MNIST dataset, the SCNNB-a acquires good test result of 93.56%, which is 0.29% higher than the SCNNB-b to reach the classification result of 93.27%. The SCNNB-a is 0.13% lower

**Table 1** Comparison classification results of our methods on MNIST, fashion-MNIST, and CIFAR10 datasets

Our methods	MNIST classification test accuracy (%)	Fashion-MNIST classification test accuracy (%)	CIFAR10 classification test accuracy (%)
SCNNB-a	99.48	93.56	85.86
SCNNB-b	99.46	93.27	82.38
SCNNB	99.54	93.69	86.69

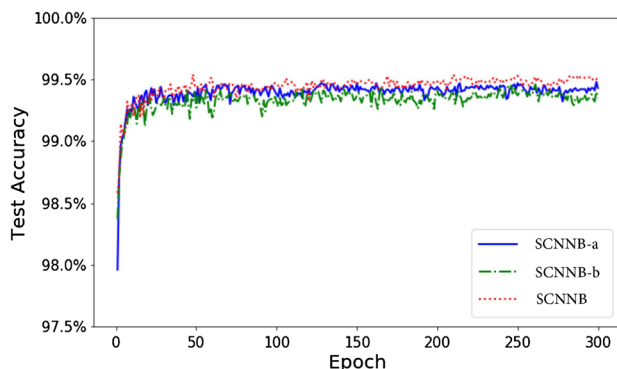
than the SCNNB (93.69%) that each convolutional layer followed by BN strategies.

The results in Table 1 show that on CIFAR10 dataset, SCNNB achieves the best classification result of 86.69%, which verifies the validity of SCNNB.

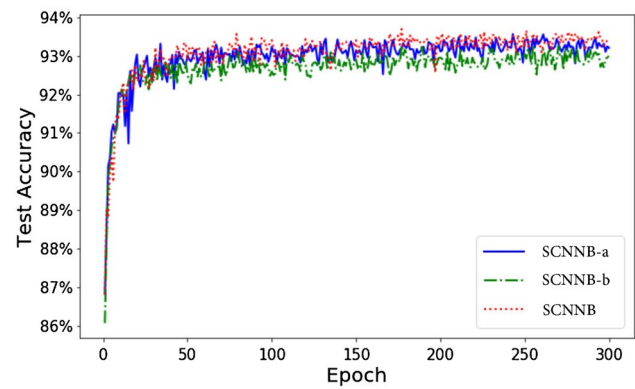
Figures 2, 3, and 4 show the test accuracy of the SCNNB, CNNB-a, and SCNNB-b on MNIST, fashion-MNIST, and CIFAR10 datasets, respectively. In Figs. 2, 3, and 4, it is obviously to see that the overall trend of test classification result of SCNNB is better than SCNNB-a and SCNNB-b. On CIFAR10, the accuracy of SCNNB is 4.72% (average value) and 3.67% (average value) higher than the classification result of SCNNB-a and SCNNB-b, respectively, which implies that the SCNNB can learn the features of data better and faster. The BN technology can speed up the convergence of the network and improve the accuracy of the model.

The comparison between SCNNB and classic deep CNN methods are shown in Table 2. On MNIST dataset, the proposed method achieves an accuracy of 99.54%, is superior to the deep CNN methods [3, 4] that include a large number of convolutional layers. The test accuracy of SCNNB is similar to the state-of-the-art deep CNN [25] on MNIST. However, the [25] network consists of  $5 \times 5$  convolution with 419/403 filters in the first / second convolutional layer and  $7 \times 7$  convolution with 288 filters in the third convolutional layer respectively. Moreover, the SCNNB network has two  $3 \times 3$  convolutions with 32 and 64 filters, respectively. Compared to these deep CNNs, the SCNNB network has smaller network structure, lower computational cost and faster training speed.

On Fashion-MNIST dataset, the test accuracy of the SCNNB is superior to many deep CNN methods [3, 13, 24]. The SCNNB is 7.26% higher than AlexNet [3], obtaining the test accuracy of 93.69%. Although the SCNNB is lower than some deep CNN methods [4, 25, 26] (3.97% lower than Zeng et al. [26] of 97.66%), the number of convolutional



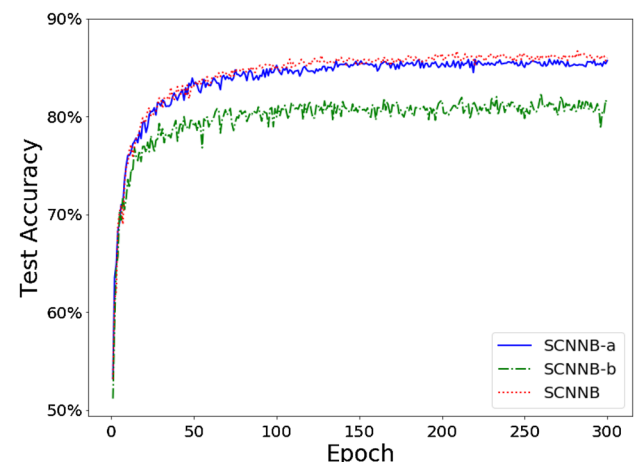
**Fig. 2** Comparison classification results of our methods on MNIST dataset



**Fig. 3** Comparison classification results of our methods on fashion-MNIST dataset

layers of deep CNN methods [4, 26] is 6 times more than the SCNNB. Compared to the SCNNB, the [15] network consists of seven convolutional layers with a large number of filters (such as  $7 \times 7/5 \times 5$  convolution with 442/382 filters). Compared to all the above deep CNN methods, the SCNNB model has a shallow and simple 4-layers network structure, which generates fewer parameters and calculations, and less time resources to train.

On the other hand, the comparison results of the SCNNB and classic shallow CNN methods on MNIST and fashion-MNIST datasets are shown in Table 3. It is obviously to see that the classification accuracy of our method is about the same as shallow CNN [18], achieving an accuracy of 99.54%. The SCNNB is only 0.12% lower than shallow CNN [20] (3 convolutional layers) that has more layers and higher time complexity on MNIST. The SCNNB with 4 layers is better than other shallow CNN methods on MNIST. The SCNNB achieves the highest classification result of 93.69%



**Fig. 4** Comparison classification results of our methods on CIFAR10 dataset

**Table 2** Comparison classification results of SCNNB and deep CNN methods on MNIST and fashion-MNIST datasets

Deep CNN methods	MNIST classification test accuracy (%)	Fashion-MNIST classification test accuracy (%)	Con./Fc number
AlexNet [3]	98.81	86.43	5/3
VGG16 [13]	–	92.89	13/3
VGG19 [13]	–	92.90	16/3
VGG16 H-CNN [24]	–	93.52	≥13/≥3
VGG19 H-CNN [24]	–	93.33	≥16/≥3
ResNet [4]	99.37	94.39	≥17/1
Ma et al. [25]	99.72	94.60	3/4 (7/4)
Zeng et al. [26]	–	97.66	13/3
SCNNB (ours)	99.54	93.69	2/2

Conv. denotes convolutional layers, and Fc is fully connected layers (including output layer). VGG16 H-CNN and VGG19 H-CNN models compose of VGG16 and VGG19 basic models and three fully-connected neural networks blocks. ResNet network has 18 layers, 34 layers, 50 layers and other network structures, and the ResNet network has the least number of layers with 18 layers. Ma et al. [25] proposes different network structure models for different image classifications. The network structure has three convolutional layers and four fully connected layers on MNIST. The network structure has 7-layer convolutional layers and 4-layer fully connected layers on fashion-MNIST. In recent years, researchers only count the layers of convolutional layers and fully connected layers when calculating network layer. We also use the same method to calculate the layers of networks

**Table 3** Comparison classification results of SCNNB and shallow CNN methods on MNIST and fashion-MNIST datasets

Shallow CNN methods	MNIST classification test accuracy (%)	Fashion-MNIST classification test accuracy (%)	Time complexity
Bhatnagar et al. [9]	–	92.54	2.0 M
Hossain et al. [16]	–	93.41	54.1 M
Petra et al. [17]	99.25	92.32	–
Agarap et al [15]	99.17	90.91	–
Poernomo et al. [18]	99.59	92.03	–
Jain et al. [20]	99.66	–	4.5 M
Gorokhovatskyi et al. [19]	98.52	–	1.2 M
Ide et al. [11]	98.89	–	–
SCNNB (ours)	99.54	93.69	3.8 M

The time complexity denotes the number of computation (theoretical time complexity), rather than the actual running time. We calculate the time complexity using Eq. (3). M is million. For instance, 3.8 M is 3.8 millions

with 3.8 M time complexity on fashion-MNIST, which is 0.28% higher than shallow CNN [16] of 93.41% that time complexity is about 14 times that of the SCNNB, and which is much higher than other shallow CNN methods.

Table 4 shows these comparison results in terms of test accuracy and training time on all datasets. It is obviously to observe that SCNNB with the fewest training time is better by a large margin in most cases.

## 5 Conclusions

In this article, a novel shallow convolutional neural network (SCNNB) with batch normalization strategy is proposed for image classification. The batch normalization strategy can accelerate the convergence speed and improve accuracy of image classification. The SCNNB model has a 4-layer simple structure with 3.8 M time complexity on the benchmark image datasets. Experiments show that the SCNNB model achieves excellent classification results than the other SCNN and VGG models. In the future, instead of the fully connected layer,  $1 \times 1$  convolution and global average pooling will be introduced to reduce the number of parameters.

**Table 4** Comparison results on MNIST, fashion-MNIST, and CIFAR10 datasets

CNN methods	MNIST accuracy/time	Fashion-MNIST accuracy/time	CIFAR10 accuracy/time
AlexNet [3]	99.38%/6.6s	91.00%/13.6s	79.12%/7.1s
VGG16 [13]	99.40%/34.2s	93.63%/46.8s	90.25%/37.7s
ResNet18 [4]	99.62%/28.5s	94.30%/38.0s	92.49%/30.0s
Hossain et al. [16]	99.49%/5.6s	91.97%/14.1s	81.3%/6.5s
Jain et al. [20]	99.46%/4.6s	92.19%/10.0s	79.12%/5.9s
SCNNB (ours)	99.54%/4.3s	93.69%/9.7s	86.69%/5.6s

We report these results including test accuracy and training time by our experiments on three image datasets

**Acknowledgements** This research was funded by the [National Natural Science Foundation of China] Grant Numbers [U170120078, 61571141, 61702120 and 61672008], the [Guangdong Provincial Key Laboratory Project] Grant Numbers [2018B030322016], the [Scientific and Technological Projects of Guangdong Province] Grant Number [2017A050501039], the [Guangdong Province General Colleges and Universities Featured Innovation] Grant Number [2015GXJK080], and the [Qingyuan Science and Technology Plan Project], the Grant Number [170809111721249, 170802171710591].

## Compliance with ethical standards

**Conflict of interest** The authors declare that they have no conflict of interest.

## References

- LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. *Proc IEEE* 86(11):2278–2324
- Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. *Science* 313(5786):504–507
- Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. Presented at NIPS. *Advances in neural information processing systems*
- He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. Presented at CVPR. *Proceedings of the IEEE conference on computer vision and pattern recognition*. (Online). <https://ieeexplore.ieee.org/document/7780459>
- Agarap AF (2017) An architecture combining convolutional neural network (CNN) and support vector machine (SVM) for image classification. *arXiv preprint arXiv:1712.03541*
- Lee TK, Baddar WJ, Kim ST, Ro YM (2018) Convolution with logarithmic filter groups for efficient shallow CNN. Presented at MMM. *International conference on multimedia modeling*
- Zhang Z, Zhang Y, Zhou Z, Luo J (2018) Boundary-based image forgery detection by fast shallow CNN. *arXiv preprint arXiv:1801.06732*
- Rayar F, Uchida S (2018) On fast sample preselection for speeding up convolutional neural network training. Presented at S+SSPR. *Joint IAPR international workshops on statistical techniques in pattern recognition (SPR) and structural and syntactic pattern recognition (SSPR)*
- Bhatnagar S, Ghosal D, Kolekar MH (2017) Classification of fashion article images using convolutional neural networks. Presented at ICIIIP. *2017 fourth international conference on image information processing (ICIIP)*
- Lu L, Yang Y, Jiang Y, Ai H, Tu W (2018) Shallow convolutional neural networks for acoustic scene classification. *Wuhan Univ J Nat Sci* 23(2):178–184
- Ide H, Kurita T (2018) Convolutional neural network with discriminant criterion for input of each neuron in output layer. Presented at ICONIP. *International conference on neural information processing*
- Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*
- Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*
- Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H (2017) Mobilenets: efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*
- Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L-C (2018) Inverted residuals and linear bottlenecks: mobile networks for classification, detection and segmentation. *arXiv preprint arXiv:1801.04381*
- Hossain MM, Talbert DA, Ghafoor SK, Kannan R (2018) FAWCA: a flexible-greedy approach to find well-tuned CNN architecture for image recognition problem
- Vidnerov P, Neruda R (2018) Asynchronous evolution of convolutional networks. *ITAT*
- Poernomo A, Kang D-K (2018) Biased dropout and crossmap dropout: learning towards effective dropout regularization in convolutional neural network. *Neural Netw* 104:60–67
- Gorokhovatskiy O, Peredrii O (2018) Shallow convolutional neural networks for pattern recognition problems. Presented at DSMP. *2018 IEEE second international conference on data stream mining and processing (DSMP)*
- Jain S, Chauhan R (2018) Recognition of handwritten digits using DNN, CNN, and RNN. Presented at ICACDS. *International conference on advances in computing and data sciences*
- Li H, Lin Z, Shen X, Brandt J, Hua G (2015) A convolutional neural network cascade for face detection. Presented at CVPR. *Proceedings of the IEEE conference on computer vision and pattern recognition*
- Niu Z, Zhou M, Wang L, Gao X, Hua G (2016) Ordinal regression with multiple output CNN for age estimation. Presented at CVPR. *Proceedings of the IEEE conference on computer vision and pattern recognition*
- He K, Sun J (2015) Convolutional neural networks at constrained time cost. Presented at CVPR. *Proceedings of the IEEE conference on computer vision and pattern recognition*

24. Seo Y, Shin K (2019) Hierarchical convolutional neural networks for fashion image classification. *Expert Syst Appl* 116:328–339
25. Ma B, Xia Y (2018) Autonomous deep learning: a genetic DCNN designer for image classification. arXiv preprint [arXiv:1807.00284](https://arxiv.org/abs/1807.00284)
26. Zeng S, Zhang B, Zhang Y, Gou J (2018) Collaboratively weighting deep and classic representation via  $l_2$  regularization for image classification. Presented at ACML. Asian conference on machine learning
27. Krizhevsky A, Hinton G et al (2009) Learning multiple layers of features from tiny images. Technical report, University of Toronto

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.