IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received March 2, 2019, accepted April 1, 2019, date of publication April 4, 2019, date of current version April 18, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2909401

Deep Learning-Aided OCR Techniques for Chinese Uppercase Characters in the Application of Internet of Things

YUE YIN[1], (Student Member, IEEE), WEI ZHANG[2,3], (Member, IEEE), SHENG HONG[1], (Student Member, IEEE), JIE YANG[1], (Member, IEEE), JIAN XIONG[1], (Member, IEEE), AND GUAN GUI[1], (Senior Member, IEEE)

[1]College of Telecommunication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
[2]School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
[3]Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, Nanjing University of Posts and Telecommunications, Nanjing 210023, China

Corresponding authors: Wei Zhang (zhangw@4njupt.edu.cn) and Guan Gui (guiguan@njupt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61701258, in part by the Jiangsu Specially Appointed Professor Grant under Grant RK002STP16001, in part by the Innovation and Entrepreneurship of Jiangsu High-Level Talent under Grant CZ0010617002, in part by the Summit of the Six Top Talents Program of Jiangsu under Grant XYDXX-010, and in part by the 1311 Talent Plan of Nanjing University of Posts and Telecommunications.

ABSTRACT Optical character recognition (OCR) has become one of the most important techniques in computer vision, given that it can easily obtain digital information from various images on the Internet of Things (IoT). However, existing OCR techniques pose a big challenge in the recognition of the Chinese uppercase characters due to their poor performance. In order to solve the problem, this paper proposes a deep learning-aided OCR technique for improving recognition accuracy. First, we generate a database of the Chinese uppercase characters to train four neural networks: a convolution neural network (CNN), a visual geometry group, a capsule network, and a residual network. Second, the four networks are tested on the generated dataset in terms of accuracy, network weight, and test time. Finally, in order to reduce test time and save computational resources, we also develop a lightweight CNN method to prune the network weight by 96.5% while reducing accuracy by no more than 1.26%.

INDEX TERMS Optical character recognition (OCR), convolution neural network (CNN), visual geometry group, residual network, capsule network, pruning network.

I. INTRODUCTION

Optical character recognition (OCR) is a technology that uses computer software to automatically recognize optical characters in the applications of internet of things (IoT). It is essentially a form of image classification. It is one of the most important techniques in computer vision and has attracted a large amount of attention across different applications [1], [2].

Wang et al. put forward a set of algorithms for license plate segmentation and recognition, and obtained high character recognition accuracy [3]. Inoue *et al*. proposed a method of combining classifiers using nonlinear discriminant analysis to improve the accuracy of hand-written character recognition [4]. Kokawa *et al*. proposed a Japanese text classification method based on the language features, which can greatly improve the accuracy of the text classification [5]. Using Poisson foil and an edge-enhanced maximum stable extreme value area for text recognition, the text area can be accurately separated from the image, which improves the reliability of recognition [6]. Text recognition can be applied not only to images, but also to a wide range of applications in video [7]. The main method of text recognition from video involves dividing the video into individual frames. When it comes to the recognition of Arabic text, the authors in [8] proposed an effective end-to-end trainable hybrid architecture. Their model is able to recognize Arabic text in high accuracy.

Existing OCR techniques perform very well in the recognition of English words as well as Arabic numerals. However, the accuracy of these techniques is not high for recognizing Chinese characters due to different language families [9]. It is hard to develop related techniques due to the fact that Chinese

The associate editor coordinating the review of this manuscript and approving it for publication was Jun Wu.

characters are very similar. As a result, conventional methods cannot achieve high recognition accuracy.

OCR is an important part of pattern recognition while deep learning has good performance in pattern recognition. Deep learning is therefore considered to be an effective method for handling big data and improving identification performance. Recently, it has been successfully applied in different applications [10], [11]. In [12], a deep learning-aided non-orthogonal multiple access (NOMA) scheme was proposed to improve achievable rate and access performance. The system uses the long short-term memory (LSTM) network for the NOMA system and trains the LSTM network with data under different channel conditions so that the proposed scheme can automatically detect channel characteristics while ensuring its robustness. The authors proposed a deep learning-aided super-resolution channel estimation and direction-of-arrival estimation technique by using a deep neural network (DNN) for both offline and online learning [13]. In addition, pilot allocation is also an important part in multiple-input multiple-output techniques. A new deep learning-based pilot design scheme was proposed in [14], which uses a multi-layer perceptron to infer the optimal pilot allocation scheme. The Internet of Things (IoT) has high requirements for energy and resource efficiency, and it is impossible to directly implement edge computing on the IoT [15], [16]. In order to improve the spectrum efficiency of the IoT, NOMA technology was introduced to solve the problem of energy-saving resource allocation, and a recurrent neural network (RNN) was introduced to optimize resource allocation [17].

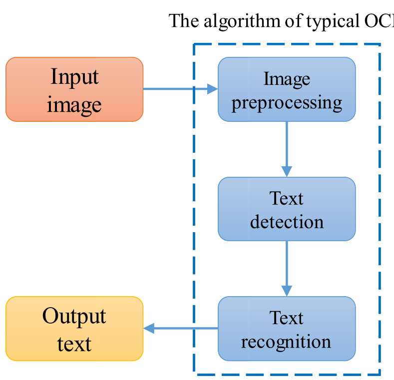

FIGURE 1. The framework of a typical OCR. Firstly, we enter the image into the system. Then, image preprocessing is performed, and the original image is subjected to tilt correction and the like. Next, locate the text area and identify the text. Finally, the system outputs the result of the text recognition.

In addition, a DNN was proposed to improve the performance of character recognition. In [18], Nawaz *et al*. created an effective hand-written dataset with a set of classes suitable for deep metric learning. Joshi *et al*. developed a deep learning-based method for improving the recognition accuracy and efficiency of hand-written Gujarati characters [19]. In [20], Rajnoha *et al*. used a deep learning method to classify hand-written characters and achieved an accuracy of 90.04%. In [21], Wiraatmaja *et al*. proposed an increased OCR technique with the aid of a deep convolutional denoising autoencoder. In [22], the proposed deep learning model achieved better accuracy on poor quality text images and an overall reduction of 21.5% in error rate compared to the existing OCR technologies.

Traditional networks of deep learning that are used to recognize characters have huge network weight and thus their computational burden is very high. To solve these problems and accelerate the development of edge computing, we consider pruning the networks. Some typical applications of deep reinforcement learning (DRL) in network slicing are described in [23], and the possible challenges of DRL in network slicing resource management are discussed. Kato et al. proposed a heterogeneous computing platform based on deep learning and intelligent routing development [24]. Compared to existing deep learning methods, the proposed methods can ensure more stable network performance when the network topology changes.

To enable the successful application of deep learning in edge computing, [25] proposed a pruning method for developing a light deep learning network. In deep networks, the output of most neurons is zero, meaning that most are useless. These inactivated neurons are often redundant and can therefore be removed without affecting the accuracy of the deep network. Re-training after network pruning can achieve the same high accuracy as the original while greatly reducing the network weight [26].

This paper proposes a lightweight neural network-based method for Chinese text recognition. The proposed method can greatly reduce network sizes and achieve high accuracy. The rest of this paper is organized as follows. Section II introduces four typical deep neural networks: the convolution neural network (CNN), visual geometry group (VGG), residual network (ResNet), and capsule network (CapsNet). Section III provides the method of Lightweight CNN and discusses its performance. Finally, Section IV provides a summary of the present paper.

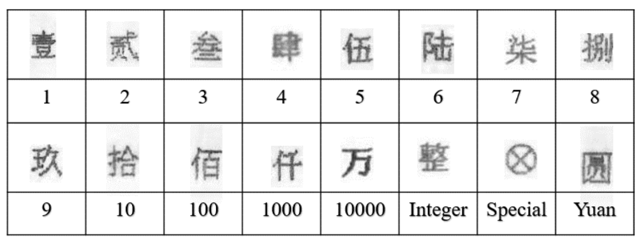

FIGURE 2. The dataset of all special Chinese uppercase characters. The penultimate pattern represents a special symbol, and the other fifteen patterns are all Chinese characters. The numbers and the English words in the second and fourth lines indicate the meaning of the corresponding Chinese characters and the special symbol.

II. NETWORKS OF DEEP LEARNING

Deep learning is an important division of machine learning and has been successfully applied in many fields such as speech recognition and natural language processing (NLP). In this section, we will train four networks to recognize

15 kinds of Chinese uppercase characters and a special symbol. Both the test set and the verification set have 480 images. We made data enhancements to the datasets, including rotating, cropping, and denoising the original image before entering the network.

A. CONVOLUTION NEURAL NETWORK

The CNN is the most famous network of deep learning and is widely applied in computer vision and NLP. It is a class of feedforward neural network with convolutional computation and deep structure [27].

For general large-scale image classification problems, a CNN can be used to construct hierarchical classifiers, and can also be used in fine classification recognition to extract discriminant features of images for other classifiers to learn. It enjoys excellent performance in image classification and has some advantages over traditional technologies like good adaptive performance and high resolution. It integrates the feature extraction function into the multi-layer perceptron via structural reorganization and reducing the weight, and omits the complicated image feature extraction process before recognition.

TABLE 1. CNN network parameters.

Layer	Output Shape	Parameter
Input	(48, 48, 1)	0
Conv2D	(44, 44, 128)	3328
Max Pooling	(22, 22, 128)	0
Conv2D	(22, 22, 64)	73792
Conv2D	(22, 22, 64)	36928
Average Pooling	(11, 11, 64)	0
Flatten	(7744)	0
Dense	(1024)	7930880
Dropout	(1024)	0
Dense	(1024)	1049600
Dropout	(1024)	0
Dense	(16)	16400

A CNN consists of three main parts. The first part is the input layer. The second part consists of a combination of convolutional layers and pooling layers. The third part consists of fully-connected layers [28]. The convolutional layer is the core component of a CNN. It consists of multiple filters. The parameters of each filter are optimized using a backpropagation algorithm. The purpose of convolution is to extract the features of the input image.

There are two main types of pooling operations. One is average pooling and the other is max pooling. The pooling layer will continuously reduce the dimensions of the data, the number of parameters, and the amount of calculation.

The detailed parameter of the CNN network is listed in Table 1. Each node of the fully connected layer is connected to all nodes of the previous layer to combine the features extracted from the front. Due to its fully connected nature, the parameters of the fully connected layer are most common. The number of parameters is 9,110,928. The weight of the CNN is 106,820 KB and its accuracy is 98.96%.

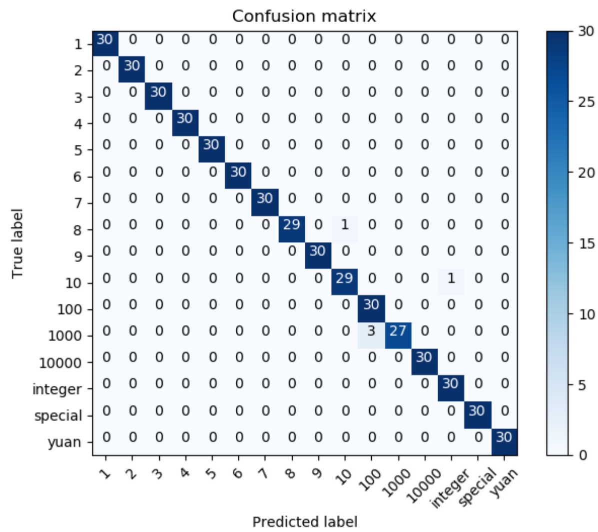

FIGURE 3. CNN confusion matrix. There are thirty images tested for each character. The horizontal axis represents the result of the system prediction, and the vertical axis represents the true label of the image.

B. VISUAL GEOMETRY GROUP

The VGG is a deep convolutional network developed by the computer vision group of Oxford University and DeepMind. A VGG and CNN are not much different in principle. According to network structure, a VGG is divided into six different classes. Each structure contains five sets of convolution layer. Each convolution layer uses 3×3 convolution kernels. Each group of convolutions are accumulated after a 2×2 maximum pooling, followed by three fully connected layers [29].

The VGG increases the depth of the neural network and uses a small convolution kernel. It has a great effect on the final classification and recognition effect of the network when it is used to capture details of images. However, the VGG consumes more computing resources and uses more parameters than a CNN, resulting in more memory usage. Most of the parameters come from the first fully connected layer of the VGG.

When training a high-level network like a VGG, we can first train the low-level network, and initialize the high-level network with the weight obtained by the former. In this way, we can accelerate the convergence of the network with less consumption of computational resources. The weight of the VGG is 157,358 KB and its accuracy is 97.92%.

C. RESIDUAL NETWORK

It is generally believed that the deeper the network layer is, the higher the features are extracted, and the better the final effect. However, the main problems encountered by deep learning for network depth are gradient disappearance and

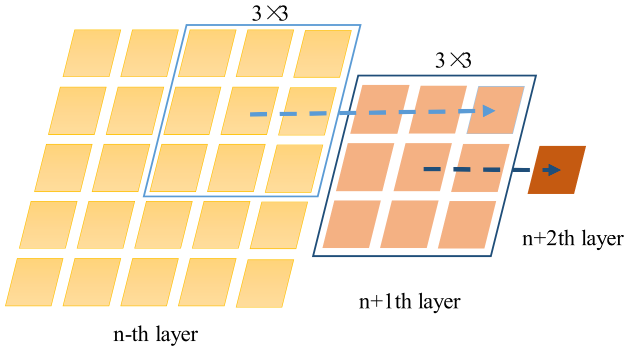

FIGURE 4. VGG implemented convolution operation with a 3 × 3 convolution kernel. Each parallelogram represents a pixel. The 5 × 5 matrix is convoluted by a 3 × 3 convolution kernel and becomes a 3 × 3 matrix. After another convolution, it becomes a 1 × 1 matrix.

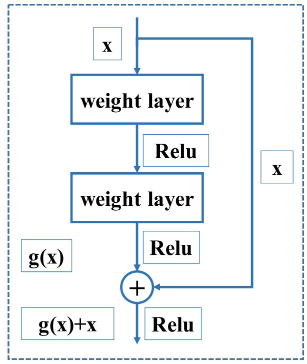

FIGURE 6. Residual unit. This residual unit often requires more than two layers, and the single-layer residual block does not improve.

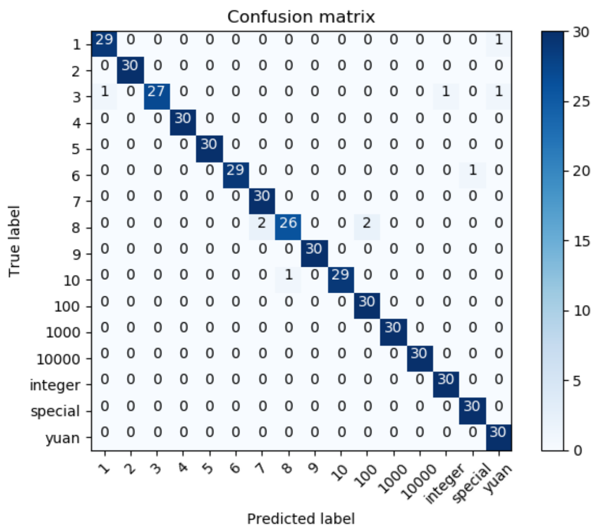

FIGURE 5. VGG confusion matrix. VGG has errors when predicting characters 1, 3, 6 and 8.

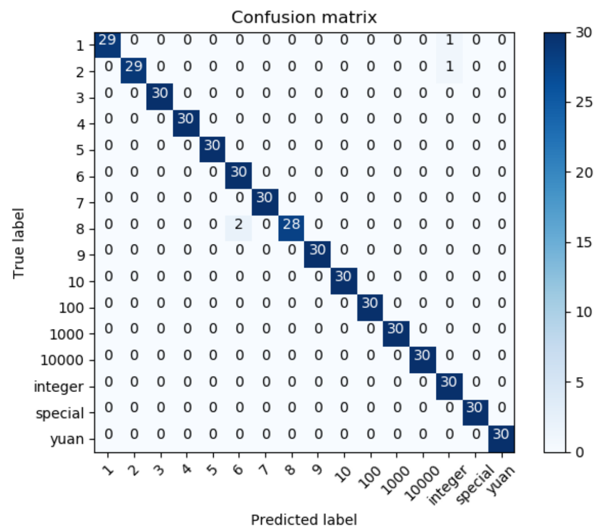

FIGURE 7. ResNet confusion matrix. ResNet has errors when predicting characters 1, 2 and 8.

gradient explosion. The traditional corresponding solution is the initialization and regularization of data. Although this solves the problem of gradient, when the depth is deepened, it will bring another problem which is the degradation of network performance. The depth is deepened, and the error rate is increased [30].

The ResNet is a new deep convolutional network proposed in 2015. Upon its birth, it won the championship of image classification, detection, and positioning in ImageNet. A ResNet is easier to optimize and can increase accuracy by adding considerable depth. The core is to solve the side effects caused by the increase in depth, which can improve network performance by simply increasing the network depth.

Some vector data of the previous layer is combined with the data that has been compressed as the following input data. Introducing more dimensional features, the network can learn more to improve accuracy. A ResNet is used to handle the degradation problem. It can solve the gradient problem and improve the performance of the network. When the performance of the ResNet reaches a bottleneck, the redundant network layer makes an identity map. The weight of the ResNet is 40,185 KB and its accuracy is 99.17%.

D. CAPSULE NETWORK

The CNN does not consider the relative positional factors of the object when identifying it. Assuming that the CNN can recognize human faces, it will give the answer of human face as long as there are two eyes, a nose, two ears, and a mouth on an image, even if they are randomly arranged. The reason is that the CNN's pooling layers reduce parameters, avoid overfitting, and discard location information [31].

A CapsNet will solve this problem by taking the location information of the object into account. It replaces the scalar output of each neuron in the CNN with a vector output. A CapsNet uses the size of the vector modulus to measure the probability of occurrence of an entity. The larger the modulus, the greater the probability. The weight of the CapsNet is 3,458 KB and its accuracy is 99.38%.

III. LIGHTWEIGHT CONVOLUTION NEURAL NETWORK

The storage and computation of neural networks on embedded devices has become a huge challenge due to storage space and power constraints. In general, the deeper the number

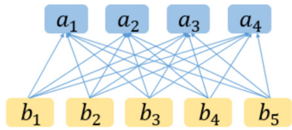

FIGURE 8. CapsNet connection relationship. Each capsule nerve unit in the previous layer is connected to each unit in the latter layer.

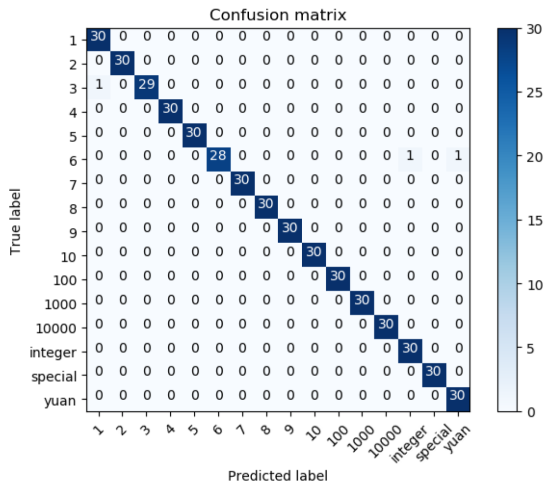

FIGURE 9. CapsNet confusion matrix. CapsNet has errors when predicting characters 3 and 6.

of layers of the neural network and the more parameters, the more accurate the conclusions are. However, accurate results mean more computing resources are consumed. For mobile devices, speed and accuracy are of equal importance. In order to solve the problem, pruning technology came into being.

Pruning is the removal of parameters that do not contribute much to the output. The neurons of the model are first sorted according to the order of their contribution to the final result. Some neurons with low contributions are then discarded so that the model runs faster and the weight file of the model is smaller. Therefore, we try to trim the neural network, remove some unnecessary network neurons, retain the weight parameters which are important to the network, and reduce the parameters in order to reduce the computational complexity of the model. If too many neurons are removed, there will be a certain loss in accuracy of the model, resulting in a large degree of decline in performance. Therefore, pruning is actually an iterative process. After deleting the neurons, we must retrain the network.

In deep neural networks, most of the neurons are activated to zero, and the neurons with 0 activation are redundant. Eliminating them can greatly reduce the model size and the energy computation. We use the Average Percentage of Zeros (ApoZ) algorithm to measure the number of values activated by 0 in each filter as a criterion for evaluating whether a filter is important.

$$APoZ_c^{(i)} = APoZ(H_c^{(i)}) = \frac{\sum_k^N \sum_j^M g(H_{c,j}^{(i)}(k) = 0)}{N \times M} \quad (1)$$

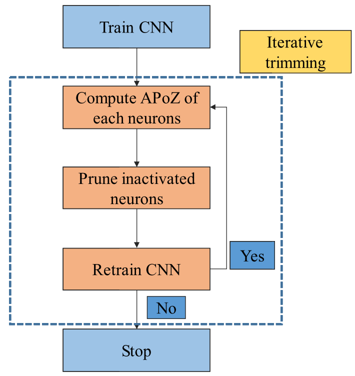

FIGURE 10. Process of the pruning network. The core of pruning is to prune neurons with APoZ = 0.

TABLE 2. Lightweight CNN network parameters.

Layer	Output Shape	Parameter
Input	(48, 48, 1)	0
Conv2D	(44, 44, 82)	2132
Max Pooling	(22, 22, 82)	0
Conv2D	(22, 22, 3)	2217
Conv2D	(22, 22, 64)	1792
Average Pooling	(10, 10, 64)	0
Flatten	(6400)	0
Dense	(39)	249639
Dropout	(39)	0
Dense	(1024)	40960
Dropout	(1024)	0
Dense	(16)	16400

where $H_c^{(i)}$ is the output of the c-th channel in the i-th layer. M represents the dimension of the output feature map of $H_c^{(i)}$, and N is the total amount of validation images. We calculate the APoZ of each neuron and then decide which neuron can be pruned.

The CNN is the most widely used neural network for deep learning, with millions of parameters and neurons. In the CNN, the vast majority of parameters are derived from the convolutional layer and the fully connected layer, where the parameters of the fully connected layer reach more than 90% of the total. However, instead of every neuron in the network coming in handy, many neurons are not activated. Cutting these inactivated neurons has less impact on the network. In practice, we can iteratively prune the CNN by synthesizing the requirements of network weights and the requirements of accuracy.

The detailed parameters of lightweight CNN is listed as in Table 2. The total number of lightweight CNN parameters is 313,140. The weight of the Lightweight CNN is 3,727 KB. Compared with the CNN, with an accuracy rate of 97.7%, the network weight is reduced by 96.5%. Also the performance comparisons of all networks are given in Table 3.

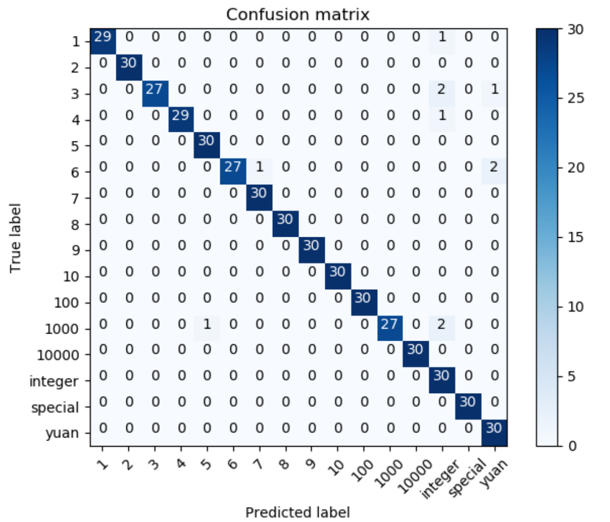

FIGURE 11. Lightweight CNN confusion matrix. In total 480 images, lightweight CNN has predicted 11 images wrong.

TABLE 3. Network comparison.

	CNN	VGG	ResNet	CapsNet	Lightweight CNN
Accuracy	98.96%	97.92%	99.17%	99.38%	97.70%
Weight (KB)	106820	157358	40185	3458	3727
Test time in CPU	1.70	4.86	47.31	0.22	1.33
Test time in GPU	1.40	0.05	12.05	0.05	0.05

IV. CONCLUSION

In this paper, we have proposed a deep learning-aided OCR technique for improving recognition accuracy for IoT applications. In order to realize this technique, we trained and tested a CNN, VGG, ResNet, and CapsNet on a dataset of Chinese uppercase characters and obtained their accuracy and test time for comparison. We also pruned the CNN by deleting the inactivated neurons of the convolution and dense layers. As a result, the network's weight dropped by 96.5%, with an accuracy of 97.70%. It is worth mentioning that the ResNet and CapsNet achieved an accuracy of 99.17% and 99.38% on the test dataset, respectively. The test time was an average of the time for each network to run 20 times on the CPU or GPU.

Applying deep learning to OCR can achieve a high accuracy and low processing time. Pruning the deep learning network can further reduce the system test time and network weight. In addition, pruning the deep network also helps drive the development of edge computing [33], [34].

REFERENCES

[1] J. Wu, K. Ota, M. Dong, and C. Li, "A hierarchical security framework for defending against sophisticated attacks on wireless sensor networks in smart cities," *IEEE Access*, vol. 4, pp. 416–424, 2016.

[2] J. Wu, S. Luo, S. Wang, and H. Wang, "NLES: A novel lifetime extension scheme for safety-critical cyber-physical systems using SDN and NFV," *IEEE Internet Things J.*, to be published. doi: 10.1109/JIOT.2018.2870294.

[3] M.-Y. Day and Y.-J. Wang, "Analysis of identifying linguistic phenomena for recognizing inference in text," in *Proc. IEEE 15th Int. Conf. Inf. Reuse Integr.*, Aug. 2014, pp. 607–612.

[4] N. Wang, X. Zhu, and J. Zhang, "License plate segmentation and recognition of Chinese vehicle based on BPNN," in *Proc. 12th Int. Conf. Comput. Intell. Secur.*, Dec. 2016, pp. 403–406.

[5] R. Inoue, H. Nakayama, and N. Kato, "Historical hand-written string recognition by non-linear discriminant analysis using kernel feature selection," in *Proc. 18th Int. Conf. Pattern Recognit.*, vol. 2, Aug. 2006, pp. 1094–1097.

[6] A. Kokawa, L. S. P. Busagala, W. Ohyama, T. Wakabayashi, and F. Kimura, "An impact of OCR on automated classification of OCR Japanese texts with parts-of-speech analysis," in *Proc. Int. Conf. Document Anal. Recognit.*, Sep. 2011, pp. 543–547.

[7] J. Mol, A. Mohammed, and B. S. Mahesh, "Text recognition using Poisson filtering and edge enhanced maximally stable extremal regions," in *Proc. Int. Conf. Intell. Comput., Instrum. Control Technol.*, Jul. 2017, pp. 302–306.

[8] S. Shetty, A. S. Devadiga, S. S. Chakkaravarthy, and K. A. V. Kumar, "Ote-OCR based text recognition and extraction from video frames," in *Proc. IEEE 8th Int. Conf. Intell. Syst. Control*, Jan. 2014, pp. 229–232.

[9] M. Jain, M. Mathew, and C. V. Jawahar, "Unconstrained scene text and video text recognition for arabic script," in *Proc. 1st Int. Workshop Arabic Script Anal. Recognit.*, Apr. 2017, pp. 26–30.

[10] Q. Li, W. An, A. Zhou, and L. Ma, "Recognition of offline handwritten Chinese characters using the Tesseract open source OCR engine," in *Proc. 8th Int. Conf. Intell. Hum.-Mach. Syst. Cybern.*, Aug. 2016, pp. 452–456.

[11] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," *China Commun.*, vol. 14, no. 11, pp. 92–111, Nov. 2017.

[12] L. F. Maimó, Á. L. P. Gómez, F. J. G. Clemente, M. G. Pérez, and G. M. Pérez, "A self-adaptive deep learning-based system for anomaly detection in 5G networks," *IEEE Access*, vol. 6, pp. 7700–7712, 2018.

[13] G. Gui, H. Huang, Y. Song, and H. Sari, "Deep learning for an effective nonorthogonal multiple access scheme," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8440–8450, Sep. 2018.

[14] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep learning for super-resolution channel estimation and doa estimation based massive MIMO system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8549–8560, Sep. 2018.

[15] K. Kim, J. Lee, and J. Choi, "Deep learning based pilot allocation scheme (DL-PAS) for 5G massive MIMO system," *IEEE Commun. Lett.*, vol. 22, no. 4, pp. 828–831, Apr. 2018.

[16] K. Miyanabe, T. G. Rodrigues, Y. Lee, H. Nishiyama, and N. Kato, "An Internet of Things traffic-based power saving scheme in cloud-radio access network," *IEEE Internet Things J.*, to be published.

[17] M. Liu, T. Song, J. Hu, H. Sari, and G. Gui, "Anti-shadowing resource allocation for general mobile cognitive radio networks," *IEEE Access*, vol. 6, pp. 5618–5632, 2018.

[18] M. Liu, T. Song, and G. Gui, "Deep cognitive perspective: Resource allocation for NOMA based heterogeneous IoT with imperfect SIC," *IEEE Internet Things J.*, to be published. doi: 10.1109/JIOT.2018.2876152.

[19] S. Nawaz, A. Calefati, N. Ahmed, and I. Gallo, "Hand written characters recognition via deep metric learning," in *Proc. 13th IAPR Int. Workshop Document Anal. Syst.*, Apr. 2018, pp. 417–422.

[20] D. S. Joshi and Y. R. Risodkar, "Deep learning based Gujarati handwritten character recognition," in *Proc. Int. Conf. Adv. Commun. Comput. Technol.*, Feb. 2018, pp. 563–566.

[21] M. Rajnoha, R. Burget, and M. K. Dutta, "Handwriting comenia script recognition with convolutional neural network," in *Proc. 40th Int. Conf. Telecommun. Signal Process.*, Jul. 2017, pp. 775–779.

[22] C. Wiraatmaja, K. Gunadi, and I. N. Sandjaja, "The application of deep convolutional denoising autoencoder for optical character recognition preprocessing," in *Proc. Int. Conf. Soft Comput., Intell. Syst. Inf. Technol.*, Sep. 2017, vol. 25, no. 3, pp. 72–77.

[23] T. C. Wei, U. U. Sheikh, and A. A.-H. A. Rahman, "Improved optical character recognition with deep neural network," in *Proc. IEEE 14th Int. Colloq. Signal Process. Appl.*, Mar. 2018, pp. 245–249.

[24] R. Li *et al.*, "Deep reinforcement learning for resource management in network slicing," *IEEE Access*, vol. 6, pp. 74429–74441, 2018.

[25] Z. M. Fadlullah, B. Mao, F. Tang, and N. Kato, "Value iteration architecture based deep learning for intelligent routing exploiting heterogeneous computing platforms," *IEEE Trans. Comput.*, to be published. doi: 10.1109/TC.2018.2874483.

[26] H. Hu, R. Peng, Y.-W. Tai, and C.-K. Tang. (2016). ''Network trimming: A data-driven neuron pruning approach towards efficient deep architectures.'' [Online]. Available: https://arxiv.org/abs/1607.03250
[27] R. Esfandiarpoor, M. Hajabdollahi, and N. Karimi, ''Simplified neural network based on auxiliary layer and adaptive pruning rate,'' in *Proc. Iranian Conf. Elect. Eng. (ICEE)*, May 2018, pp. 1–5.
[28] Y. LeCun, Y. Bengio, and G. Hinton, ''Deep learning,'' *Nature*, vol. 521, pp. 436–444, May 2015.
[29] S. P. Kannojia and G. Jaiswal, ''Ensemble of hybrid CNN-ELM model for image classification,'' in *Proc. 5th Int. Conf. Signal Process. Integr. Netw.*, Feb. 2018, pp. 538–541.
[30] K. Simonyan and A. Zisserman. (2015). ''Very deep convolutional networks for large-scale image recognition.'' [Online]. Available: https://arxiv.org/abs/1409.1556
[31] K. He, X. Zhang, S. Ren, and J. Sun, ''Deep residual learning for image recognition,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[32] C. Xiang, L. Zhang, Y. Tang, W. Zou, and C. Xu, ''MS-CapsNet: A novel multi-scale capsule network,'' *IEEE Signal Process. Lett.*, vol. 25, no. 12, pp. 1850–1854, Dec. 2018.
[33] J. Wu, M. Dong, K. Ota, J. Li, and Z. Guan, ''Big data analysis-based secure cluster management for optimized control plane in software-defined networks,'' *IEEE Trans. Netw. Service Manag.*, vol. 15, no. 1, pp. 27–38, Mar. 2018.
[34] J. Wu, M. Dong, K. Ota, J. Li, and Z. Guan, ''FCSS: Fog computing based content-aware filtering for security services in information centric social networks,'' *IEEE Trans. Emerg. Topics Comput.*, to be publihsed. doi: 10.1109/TETC.2017.2747158.

YUE YIN (S'18) is currently pursuing the master's degree in communication engineering with the Nanjing University of Posts and Telecommunications, Nanjing, China. Her research interest includes deep learning for the Internet of Things.

WEI ZHANG (M'19) received the Dr.Eng. degree in computer science from Soochow University, Suzhou, China, in 2008. From 2008 to 2011, he was Postdoctoral Research Fellow with the Nanjing University of Posts and Telecommunications, Nanjing, China. In 2016, he was a Visiting Scholar with Purdue University. Since 2012, he has been a Professor with the School of Computer Science, Nanjing University of Posts and Telecommunications. His current research interests include computer version and machine learning.

SHENG HONG (S'18) is currently pursuing the master's degree in communication engineering with the Nanjing University of Posts and Telecommunications, Nanjing China. His research interest includes deep learning for the Internet of Things.

JIE YANG (M'18) received the B.Sc., M.Sc., and Ph.D. degrees in communication engineering from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2003, 2006, and 2018, respectively, where she is currently an Assistant Professor.

JIAN XIONG (M'17) received the B.Sc. degree in computer science and technology from the Anhui University of Finance and Economics, Bengbu, China, in 2007, the M.Sc. degree in computer application technology from Xihua University, Chengdu, China, in 2010, and the Ph.D. degree in single and information processing from the University of Electronic Science and Technology of China, Chengdu, in 2015. In 2014, he was a Research Assistant with the Image and Video Processing Laboratory, The Chinese University of Hong Kong, Hong Kong. Since 2015, he has been a Lecturer with the College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing, China. His current research interests include image and video coding, pattern recognition, and machine learning.

GUAN GUI received the Dr.Eng. degree in information and communication engineering from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2012.

From 2009 to 2014, he was a Research Assistant and a Postdoctoral Research Fellow with the Wireless Signal Processing and Network Laboratory (Prof. Fumiyuki Adachi laboratory), Department of Communications Engineering, Graduate School of Engineering, Tohoku University. From 2014 to 2015, he was an Assistant Professor with the Department of Electronics and Information System, Akita Prefectural University. Since 2015, he has been a Professor with the Nanjing University of Posts and Telecommunications (NJUPT), Nanjing, China. He has published more than 200 international peer-reviewed journal/conference papers. He is currently engaged in the research of deep learning, compressive sensing, and advanced wireless techniques. He received the Member and Global Activities Contributions Award at the IEEE ComSoc and seven Best Paper Awards: ICC2014, VTC2014-Spring, ICC2017, ICNC2018, ADHIP2018, CSPS2018, and ICEICT2019. He was selected as the Jiangsu Specially-Appointed Professor, in 2016, the Jiangsu High-level Innovation and Entrepreneurial Talent, in 2016, the 1311 Talent Plan of NJUPT, in 2017, the Jiangsu Six Top Talent, in 2018, and the Nanjing Youth Award, in 2018. He has been an Editor of *Security and Communication Networks*, from 2012 to 2016, the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, since 2017, *KSII Transactions on Internet and Information Systems*, since 2017, IEEE ACCESS, since 2018, and *Journal of Communication*, since 2019, and the Editor-in-Chief of the *EAI Transactions on Artificial Intelligence*, since 2018.