A Novel Image Classification Method with CNN-XGBoost Model

Xudie Ren, Haonan Guo, Shenghong Li(✉), Shilin Wang, and Jianhua Li

School of Cyber Space Security, Shanghai Jiao Tong University, 800 Dong Chuan Road, Shanghai 200240, China
{renxudie, haonan2012, shli, wsl, lijh888}@sjtu.edu.cn

Abstract. Image classification problem is one of most important research directions in image processing and has become the focus of research in many years due to its diversity and complexity of image information. In view of the existing image classification models' failure to fully utilize the information of images, this paper proposes a novel image classification method of combining the Convolutional Neural Network (CNN) and eXtreme Gradient Boosting (XGBoost), which are two outstanding classifiers. The presented CNN-XGBoost model provides more precise output by integrating CNN as a trainable feature extractor to automatically obtain features from input and XGBoost as a recognizer in the top level of the network to produce results. Experiments are implemented on the well-known MNIST and CIFAR-10 databases. The results prove that the new method performs better compared with other methods on the same databases, which verify the effectiveness of the proposed method in image classification problem.

Keywords: Convolutional Neural Network · eXtreme Gradient Boosting · Image classification

1 Introduction

Image classification problem is one of the key research objectives in the field of image processing, and has a wide range of applications in object recognition, content understanding as well as image matching and so on. Over the years, although there have been substantial research results, classification problem is still the focus of researches due to the complexity and diversity of image information. This task worth researching for it can be wildly applied in the fields of pattern recognition and computer vision. Image classification methods vary from numerous aspects including Support Vector Machine (SVM), Nearest Neighbor (NN), Gradient Boosting (GB), Convolutional Neural Network (CNN), etc. These machine learning and data-driven algorithms are able to efficiently classify images and have indicated their reliability and validity. Nevertheless, most of the existing methods on the application of image classification do not sufficiently utilize image information as well as establish robust features for recognizing, thus leaving the space for improve the recognition rate through providing reliable high-level features.

C. Kraetzer et al. (Eds.): IWDW 2017, LNCS 10431, pp. 378–390, 2017.
DOI: 10.1007/978-3-319-64185-0_28

In the aspect of intelligent image classification methods, the model based on neural network is one of the important research directions. Deep neural networks can theoretically approximate any complex function and effectively solve the problems of image feature extraction and classification. However, due to the model complexity, training difficulty and high cost, this kind of structure can hardly obtain very effective application. Recently, the latest research on neural network- Deep Learning techniques have obtained continuous breakthroughs and developments in many fields, including image classification [1, 2], object detection [3, 4] and face recognition [5–7], etc. Deep architectures have been successfully applied to large-scale image processing system and achieved the state of the art performance, which showing an optimistic prospect to solve the classification task.

Feature extraction is the most important process in an automatic image classification system. The feature quality can directly influence the recognition performance which leads to a time-consuming feature engineering in traditional image classification task. CNN is an efficient Deep Learning model with hierarchical structure to learn high quality features at each layer. Since the model can reduce the complexity of network structure and the number of parameters through local receptive fields, weight sharing and pooling operation, it has been widely used in image classification problem and achieved excellent results. It also can directly input raw images in automatic classification system, which contributes to save more image information for the following feature extraction.

On the other hand, classification is another significant process in an automatic image classification system. CNN has been recognized as the most powerful and effective mechanism for feature extraction, but traditional classifiers connected to CNN do not fully understand the extracted features. Therefore showing a promising direction for proposing new solution to the image classification problem. eXtreme Gradient Boosting (XGBoost) [9] is an integrated learning algorithm based on GB, the principle of which is to achieve accurate classification results through iterative computation of weak classifiers. XGBoost is widely applied in many domains [7, 8, 10] because of its high efficiency and accuracy.

Motivated by the above facts, this paper explores the incorporation of CNN model and XGBoost algorithm since both CNN and XGBoost have already perform excellently in image classification problem. A novel image classification method with CNN-XGBoost model is proposed to improve the performance of image classification problem. The proposed CNN-XGBoost model provides more precise output by integrating CNN as a trainable feature extractor to automatically obtain features from input and XGBoost as a recognizer in the top level of the network to produce results. Such unique two-stage model guarantees the high reliability feature extraction and classification.

The rest of the paper is organized as follows: Sect. 2 introduces the basic concepts of CNN and XGBoost respectively. Then the CNN-XGBoost model is also described in this section. Experimental results on the well-known MNIST and CIFAR-10 databases are presented and discussed in Sect. 3. Finally, Sect. 4 concludes the paper.

2 The Novel CNN-XGBoost Model

In this paper, we propose a novel image classification method with CNN-XGBoost model to improve the classification performance. By integrating CNN as a trainable feature extractor to automatically obtain features from input and XGBoost as a recognizer in the top level of the network to produce results, our method can guarantee the high reliability feature extraction and classification. In the following section, we will briefly describe the two brilliant classifiers respectively and introduce the novel CNN-XGBoost model at last.

2.1 Convolutional Neural Network

Convolutional neural network (CNN) was first proposed by Professor Yann LeCun and his colleagues at the University of Toronto in Canada and used for recognition and classification of handwriting digital images [11]. CNN takes advantage of the concepts of receptive fields, weight sharing and sub-sampling (pooling) to reduce the complexity of the network structure and the number of parameters. "Receptive field" is equivalent to constructing a number of spatially localized filters which can obtain some salient features of the input. While "weight sharing" can reduce the number of parameters which needs to be trained. "Pooling" can simplify the model and prevent it from over-fitting.

A typical CNN is consisted of alternating convolution and sub-sampling layers, then turns into fully connected layers when approaching to the last output layer. It usually adjusts all the filter kernels (convolution kernels) by back-propagation algorithm [12], which is based on stochastic gradient descent algorithm, to reduce the gap between the network output and the training labels. Overall, the convolution layer (C layer) obtains the local features by connected with local receptive fields. The sub-sampling layer (S layer) is a mapping feature layer which is used for pooling operation and completing the secondary extraction calculations. Each C layer is followed by an S layer, and the special twice feature extraction structure makes convolutional neural network have strong distortion tolerance on the input images.

LeNet-5 [11] is a classic convolutional neural network architecture for handwritten digit recognition proposed by Yann LeCun et al. The structure consists of 8 layers: an input layer, two C layers, two S layers and three fully connected layers and an output layer. In order to reduce the computational complexity, the specific structure of CNN for image classification in this paper is a simplified version of LeNet-5, which includes an input layer (Input), two C layers (C1, C2), two S layers (S1, S2), fully connected layers and an output layer (Output). Figure 1 demonstrates the specific structure of CNN for image classification. Since a convolution kernel of the convolution layer can only extract one characteristic of input feature maps, it requires multiple convolution kernels to extract different features.

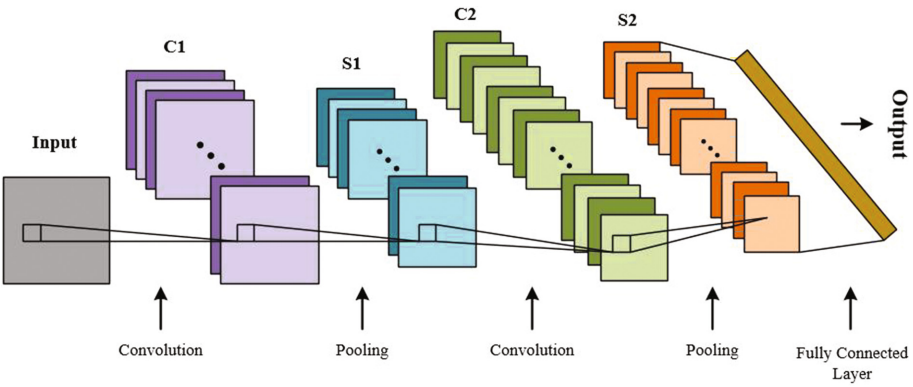

Fig. 1. The Specific structure of CNN for image classification

2.2 eXtreme Gradient Boosting

XGBoost has been widely used in many fields to achieve state-of-the-art results on many data challenges, which is a high effective scalable machine learning system for tree boosting. Developed by Tianqi Chen et al. the scalability in all scenarios of XGBoost is due to several important systems and algorithmic optimizations, which includes a novel tree learning algorithm, a theoretically justified weighted quantile sketch procedure as well as parallel and distributed computing [13].

Tree boosting is a very effective ensemble learning algorithm, which can transform several weak classifiers into a strong classifier for better classification performance. Let $D = \{(x_i, y_i)\}(|D| = n, x_i \in \mathbb{R}^m, y_i \in \mathbb{R}^n)$ represents a database with n examples and m features. A tree boosting model output $\hat{y}_i$ with K trees is defined as follows:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F \tag{1}$$

where $F = \{f(x) = \omega_q(x)\}(q : \mathbb{R}^m \to T, \omega \in \mathbb{R}^T)$ is the space of regression or classification trees (also known as CART). Each f_k divides a tree into structure part q and leaf weights part ω. Here T denotes the number of leaves in the tree.

The set of function f_k in the tree model can be learned by minimizing the following objective function:

$$O = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \tag{2}$$

The first term l in Eq. (2) is a training loss function which measures the distance between the prediction $\hat{y}_i$ and the object y_i. The second term Ω in Eq. (2) represents the penalty term of the tree model complexity.

Tree boosting model whose objective function is Eq. (2) cannot be optimized through traditional optimization methods in Euclidean space. Gradient Tree Boosting is

an improved version of tree boosting by training tree model in an additive manner, which means the prediction of the t-th iteration $\hat{y}^{(t)} = \hat{y}^{(t-1)} + f_t(x)$. And the objective function in t-th iteration is changed as:

$$O^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \tag{3}$$

XGBoost approximates Eq. (3) by utilizing the second order Taylor expansion and the final objective function at step t can be rewritten as:

$$O^{(t)} \simeq \tilde{O}^{(t)} = \sum_{i=1}^{n} [l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_t) \tag{4}$$

where g_i and h_i are first and second order gradient statistics on the loss function, and $\Omega(f) = \gamma T + \frac{1}{2}\lambda \|\omega\|^2$ in XGBoost.

Denote $I_j = \{i | q(x_i) = j\}$ as the instance set of leaf j, after removing the constant terms and expanding Ω, Eq. (4) can be simplified as:

$$\tilde{O}^{(t)} = \sum_{j=1}^{T} [(\sum_{i \in I_j} g_i)\omega_j + \frac{1}{2}(\sum_{i \in I_j} h_i + \lambda)\omega_j^2] + \gamma T \tag{5}$$

The solution weight ω_j^* of leaf j for a fixed tree structure $q(x)$ can be obtained by applying the following equation:

$$\omega_j^* = -\frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda} \tag{6}$$

After substituting ω_j^* into Eq. (5), there exists:

$$\tilde{O}(q) = -\frac{1}{2} \sum_{j=1}^{T} \frac{(\sum_{i \in I_j} g_i)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T \tag{7}$$

Define Eq. (7) as a scoring function to evaluate the tree structure $q(x)$ and find the optimal tree structures for classification. However, it is impossible to search the whole possible tree structures q in practice. [13] describes a greedy algorithm that starts from a single leaf and iteratively adds branches to grow the tree structure. Whether adding a split to the existing tree structure can be decided by the following function:

$$O_{split} = \frac{1}{2} \left[\frac{(\sum_{i \in I_L} g_i)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{(\sum_{i \in I_R} g_i)^2}{\sum_{i \in I_R} h_i + \lambda} - \frac{(\sum_{i \in I} g_i)^2}{\sum_{i \in I} h_i + \lambda} \right] - \gamma \tag{8}$$

where I_L and I_R are the instance sets of left and right nodes after the split and $I = I_L \cup I_R$.

XGBoost is a fast implementation of GB algorithm, which has the advantages of fast speed and high accuracy. This XGBoost classifier is added to the top level of the CNN to produce results for image classification in our paper.

2.3 CNN-XGBoost Model

In this paper, the specific structure of the CNN-XGBoost model for image classification is shown in Fig. 2. First, the input image data is normalized and transfer to the input layer of CNN. After training CNN by BP algorithm for several epochs to obtain a proper structure for image classification, XGBoost replaces the output layer, a soft-max classifier, of CNN and utilizes the trainable features from CNN for training. Finally, the CNN-XGBoost model gets the new classification results of testing images. Our CNN-XGBoost model can automatically obtain features from input and provides more precise classification results combining the two outstanding classifiers.

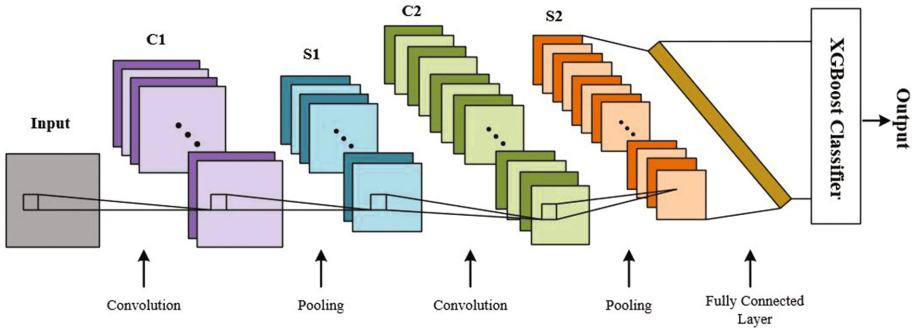

Fig. 2. The Specific structure of CNN-XGBoost model for image classification

3 Experimental Results

In order to verify the improvement and validity of the above-mentioned method, we compare it with the classical ones by carrying out experiments on different databases respectively. We also compare the classification results of different methods on the same databases to evaluate the effectiveness of CNN-XGBoost model. The databases, parameter settings and classification results are shown in the following.

3.1 Database

The two selected databases in this paper are two commonly databases used in image classification problems. They are MNIST handwritten digital database and CIFAR-10 color image database. The two databases are universal, which means it is convenient to compare to other methods.

MNIST handwritten digital database is a subset of NIST dataset, which is composed of SD-1 and SD-3 dataset. It has a total number of sixty thousand training pictures and ten thousand test images, all of which are handwritten 0-9 grayscale image.

Sixty thousand training samples are handwritten digits from about 250 individuals, part of the dataset is displayed in Fig. 3.

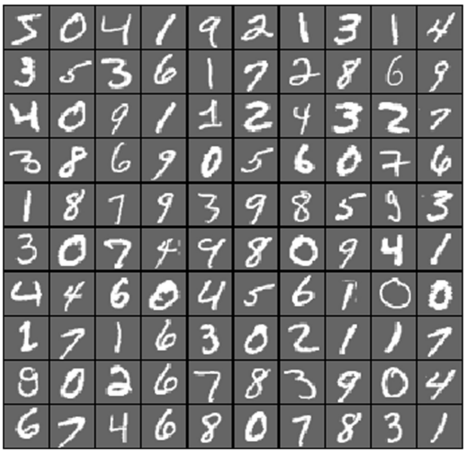

Fig. 3. Part of MNIST database

CIFAR-10 database contains 10 categories with a total of 60000 color pictures. It is divided into five training sets and a sample set and each set are ten thousand pictures. Figure 4 is the results for random selection of 10 picture results.

Fig. 4. Random selection of Cifar-10 database

3.2 Parameter Settings

To determine the parameters of CNN based on PCA initialization for image classification, we use 5 iterations as standard and calculate average classification accuracy rates of 5 times for diverse numbers of convolution kernels of the two C layers on the two databases. Finding the numbers of feature maps corresponding to maximum accuracy rate as parameters.

Considering the complexity and run-time of the network structure, we have the numbers of the first C layer feature maps tuned in the range from 3 to 12 with step size of 1, and the numbers of the second C layer feature maps in the range from 3 to 21 with step size of 3 for MNIST database. Similarly, we experiencedly have the numbers of the first C layer feature maps tuned in the range from 22 to 40 with step size of 2, and the numbers of the second C layer feature maps in the range from 24 to 72 with step size of 8 for CIFAR-10 database. Then we calculate the CNN classification accuracy rates respectively for each different feature maps. As it is shown in Figs. 5 and 6, the numbers of the feature maps in MNIST database are chose as 10 and 21, while CIFAR-10 database are 38 and 72.

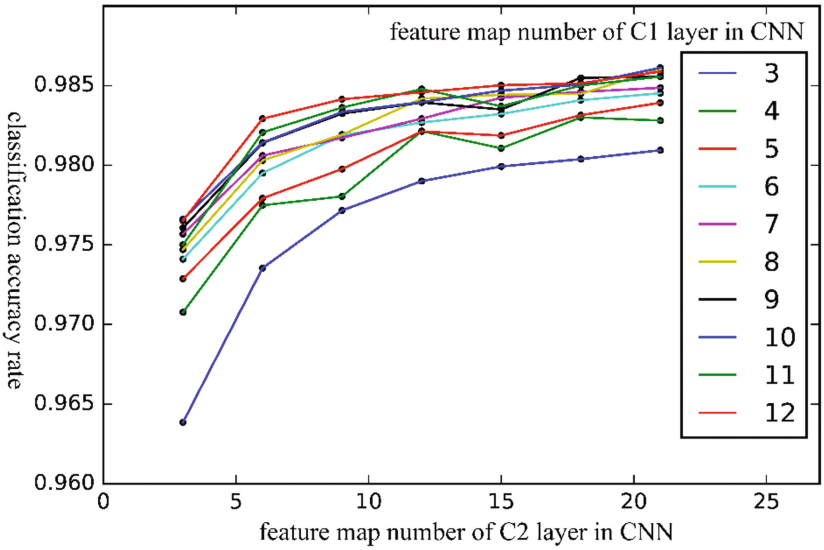

Fig. 5. Feature map number selection on MNIST database

Taking into account that the image sizes of MNIST and CIFAR-10 datasets are 28×28 and 32×32 respectively, the size of convolution kernels and sampling area are set on the basis of LeNet-5 [11] as 5×5 and 2×2.

When training the CNN on the above two databases, we take 128 pictures as a batch and whole pictures as an iteration to calculate the classification accuracy for 100 iterations after determining the feature map numbers. The classification accuracy rates on train and test sets of the two databases are displayed in Figs. 7 and 8 respectively.

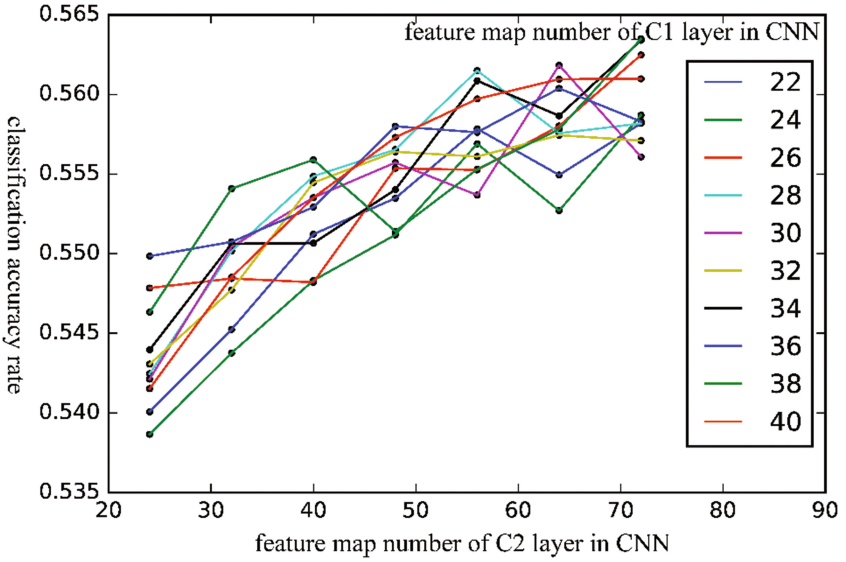

Fig. 6. Feature map number selection on CIFAR-10 database

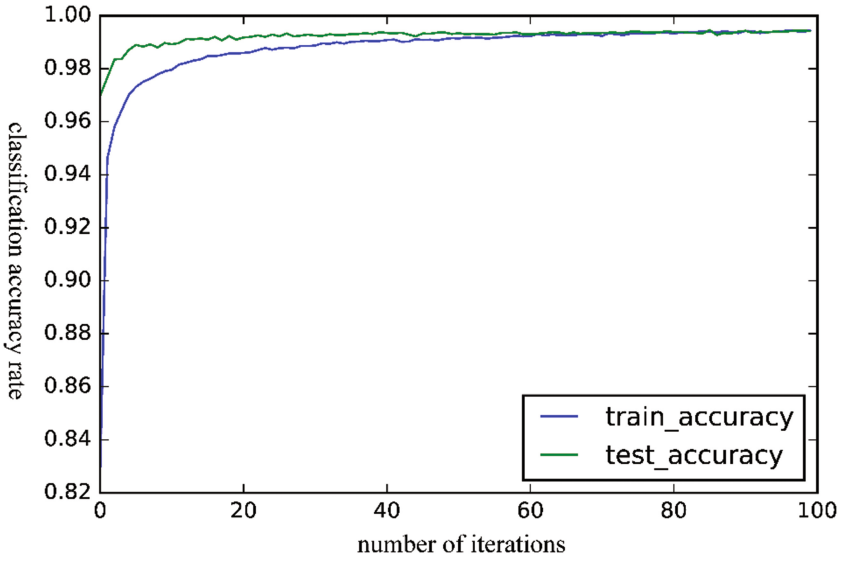

Fig. 7. Classification accuracy rates of MNIST database

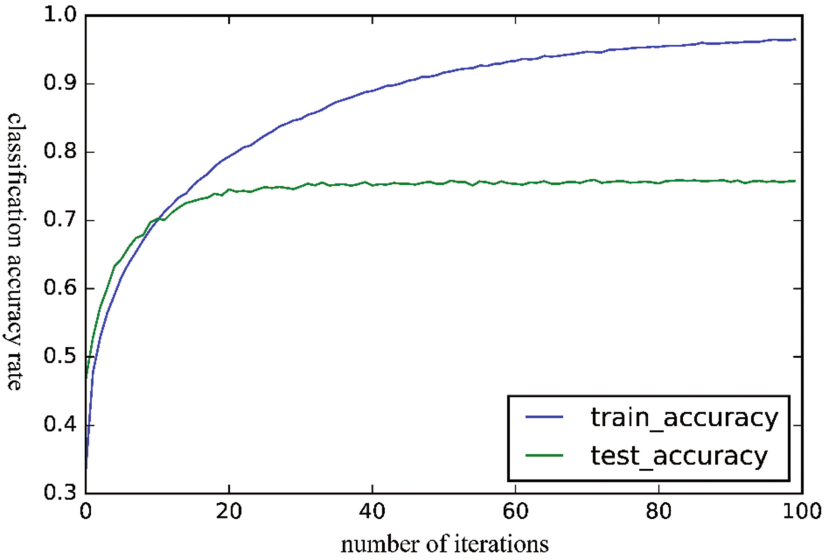

Fig. 8. Classification accuracy rates of Cifar-10 database

For the selection of parameters in XGBoost classifier, it can be determined by testing the final classification accuracy rate under different iterations, maximum tree depths and shrinkage steps. In this paper, we choose five iteration values of 100, 200, 300, 400, 500, five maximum tree depth values of 4, 6, 8, 10, 12 and five shrinkage step values of 0.005, 0.01, 0.05, 0.1, 0.2 to enumerate the classification accuracy rates in different XGBoost parameter settings and the final details of CNN-XGBoost model parameter settings are described in Table 1.

Table 1. Parameters in CNN-XGBoost model

	Parameter		MNIST	CIFAR-10
CNN	Learning rate		0.01	
	Kernel size		5	
	Pooling size		2	
	Feature map numbers	C1 Layer	10	38
		C2 Layer	21	72
XGBoost	Iterations		400	300
	Maximum tree depth		12	12
	Shrinkage step		0.05	0.01

3.3 Results and Analysis

To test the effectiveness and reality of the proposed model in this paper for image classification, we evaluate it on the above two database. In addition, we compare the classification accuracy rate with several different intelligent classification models on the

same databases and take 100 iterations as a uniform standard for the trainable models. SAE is the abbreviation for Stacked Auto-Encode. Table 2 lists the exact accuracy rates of these models. Evidently, it can be seen that the model proposed in this paper has higher classification accuracy rates on two databases than others and presents that the model can really improve the classification performance. The reason probably lies in that we utilize CNN to automatically extract high quality features with less loss of image information and high speed XGBoost to achieve efficient classification. These observations impressively demonstrate the effectiveness and reality of the proposed novel image classification method with CNN-XGBoost model. If we continue to increase the number of iterations of CNN optimization process and improve the hardware operating conditions, we believe it can get higher classification accuracy.

Table 2. Classification accuracy rate of several difference models on two databases

Database	Model	Accuracy rate (%)
MNIST	K-means [14]	95.00
	Linear SVM [15]	98.90
	SAE+CNN [16]	98.84
	CNN	98.80
	CNN-SVM [17]	99.15
	CNN-XGBoost	99.22
CIFAR-10	LR on Raw Pixels [18]	37.32
	Linear SVM [18]	42.30
	SAE+CNN [16]	62.74
	CNN	76.28
	CNN-SVM [17]	78.06
	CNN-XGBoost	80.77

4 Conclusion

In order to further enhance the classification performance of a typical Deep Learning framework – CNN, which owns outstanding performance in the image classification problem, this paper proposes novel image classification method with CNN-XGBoost model. By combining the CNN and XGBoost classifiers, this model provides more precise output by integrating CNN as a trainable feature extractor to automatically obtain features from input and XGBoost as a recognizer in the top level of the network to produce results. Experiments are implemented on the well-known MNIST and CIFAR-10 databases to examine the performance and the results demonstrate that the new method performs better compared with other methods on the same databases, which verify the effectiveness and reality of the proposed method in image classification problem.

Further work will be focus on adjusting the CNN structure to further extract higher quality features and speeding up the convergence of the cost function by changing the optimization techniques to promote the classification result and training effect.

Acknowledgement. This research work is funded by the National Key Research and Development Project of China (2016YFB0801003), the Key Laboratory for Shanghai Integrated Information Security Management Technology Research.

References

1. Hinton, G.E., Osindero, S., Teh, Y.W.: A fast learning algorithm for deep belief nets. Neural Comput. **18**(7), 1527–1554 (2006)
2. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, vol. 25(2), pp. 1097–1105 (2012)
3. Erhan, D., Szegedy, C., Toshev, A., et al.: Scalable object detection using deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2147–2154 (2014)
4. Diao, W., Sun, X., Zheng, X., et al.: Efficient saliency-based object detection in remote sensing images using deep belief networks. IEEE Geosci. Remote Sens. Lett. **13**(2), 137–141 (2016)
5. Li, C., Wei, W., Wang, J., Tang, W., Zhao, S.: Face recognition based on deep belief network combined with center-symmetric local binary pattern. In: Park, J., Jin, H., Jeong, Y. S., Khan, M. (eds.) Advanced Multimedia and Ubiquitous Engineering. LNEE, vol. 393, pp. 277–283. Springer, Singapore (2016). doi:10.1007/978-981-10-1536-6_37
6. Taigman, Y., Yang, M., Ranzato, M.A., et al.: Deepface: closing the gap to human-level performance in face verification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1701–1708 (2014)
7. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: a unified embedding for face recognition and clustering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 815–823 (2015)
8. Song, R., Chen, S., Deng, B., Li, L.: eXtreme gradient boosting for identifying individual users across different digital devices. In: Cui, B., Zhang, N., Xu, J., Lian, X., Liu, D. (eds.) WAIM 2016, Part I. LNCS, vol. 9658, pp. 43–54. Springer, Cham (2016). doi:10.1007/978-3-319-39937-9_4
9. Mackey, L., Bryan, J., Mo, M.Y.: Weighted classification cascades for optimizing discovery significance in the HIGGSML challenge. In: NIPS 2014 Workshop on High-energy Physics and Machine Learning, pp. 129–134 (2015)
10. Bekkerman, R.: The present and the future of the KDD cup competition: an outsider's perspective
11. Lecun, Y., Boser, B., Denker, J.S., et al.: Handwritten digit recognition with a back-propagation network. In: Advances in Neural Information Processing Systems, pp. 396–404 (1990)
12. Zipser, D., Andersen, R.A.: A back-propagation programmed network that simulates response properties of a subset of posterior parietal neurons. Nature **331**(6158), 679–684 (1988)
13. Chen, T., Guestrin, C.: Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794. ACM (2016)
14. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)

15. Chan, T.H., Jia, K., Gao, S., et al.: Pcanet: a simple deep learning baseline for image classification. IEEE Trans. Image Process. **24**(12), 5017–5032 (2015)
16. Ao, D.: Integration of Unsupervised Feature Learning and Neural Networks Applied to Image Recognition, pp. 19–37. South China University of Technology (2014)
17. Elleuch, M., Maalej, R., Kherallah, M.: A new design based-SVM of the CNN classifier architecture with dropout for offline arabic handwritten recognition. Procedia Comput. Sci. **80**, 1712–1723 (2016)
18. Le, Q., Sarlós, T., Smola, A.: Fastfood-approximating kernel expansions in loglinear time. In: Proceedings of the International Conference on Machine Learning (2013)
19. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006)
20. Krizhevsky, A.: Learning multiple layers of features from tiny images. Master's thesis, pp. 30–35. University of Toronto (2009)