Research Article
A Deep Convolutional Neural Network Model for Intelligent Discrimination between Coal and Rocks in Coal Mining Face

Lei Si,1,2 Xiangxiang Xiong,1 Zhongbin Wang,1,2 and Chao Tan1

1School of Mechatronic Engineering, China University of Mining & Technology, No. 1 Daxue Road, Xuzhou 221116, China
2Jiangsu Engineering Technology Research Center on Intelligent Equipment for Fully Mining and Excavating, Xuzhou 221116, China

Correspondence should be addressed to Xiangxiang Xiong; 15675114561@163.com

Received 10 September 2019; Revised 19 December 2019; Accepted 9 January 2020; Published 14 March 2020

Abstract

Accurate identification of the distribution of coal seam is a prerequisite for realizing intelligent mining of shearer. This paper presents a novel method for identifying coal and rock based on a deep convolutional neural network (CNN). Three regularization methods are introduced in this paper to solve the overfitting problem of CNN and speed up the convergence: dropout, weight regularization, and batch normalization. Then the coal-rock image information is enriched by means of data augmentation, which significantly improves the performance. The shearer cutting coal-rock experiment system is designed to collect more real coal-rock images, and some experiments are provided. The experiment results indicate that the network we designed has better performance in identifying the coal-rock images.

1. Introduction

As one piece of the key equipment for comprehensive mechanized coal mining face, shearer is an essential guarantee for high production and high efficiency of coal mining [1]. In order to improve the intelligence level of shearer, it is necessary to adaptively adjust the drum height and traction speed according to the distribution of coal and rock during the coal mining process. However, accurately and quickly identifying the distribution of coal and rock in the coal seam is still a recognized technical problem in the field of comprehensive mining technology, which is becoming a major technical bottleneck restricting the intelligence level of shearer.

In recent decades, domestic and foreign scholars and research institutes have been trying to determine the distribution of coal and rock by using vibration and sound signals to control the shearer height and traction speed [2, 3]. Because of the strong noise and vibration of the coal mining face, the predicted distribution of coal and rock is not accurate and the practical application effect based on above methods is unsatisfactory and unacceptable. Image processing methods have been widely used in many fields, such as face recognition, video surveillance analysis, intelligent driving, industrial visual inspection, and text recognition [4–10]. In the coal mining face, the monitoring images of the coal seam contain abundant coal-rock feature information. Therefore, the analysis of coal-rock images is essential for the accurate discrimination between coal and rock.

In recent years, many scholars have focused on the coal-rock identification by using image processing methods. In [11], Dong and Zhao presented an improved Canny edge detection algorithm to extract the edge features of coal-rock images and then distinguished the coal and rock according to the edge features. In [12], Xue et al. used a grey level histogram of coal-rock image to identify the coal and rock. In [13], a naive Bayesian classifier was constructed based on the multiwavelet transform of coal-rock images to achieve the recognition of coal and rock. In [14], Sun and Yang proposed a coal-rock image feature extraction and recognition method based on a binary cross-diagonal texture matrix. In [15], the edge features of coal-rock images were extracted based on curve transformation and the support
A convolutional neural net-

matic operation of point-multiplication summation of two-pixel matrices, where one matrix is the input data matrix and the other matrix is the filter (convolution kernel or feature matrix).

Based on the above reasons, this paper focuses on the deep convolutional neural network to achieve the discrimination between coal and rocks. We collect coal-rock data through some coal-rock cutting experiments and enrich the data for CNN training to improve the network identification accuracy. In order to improve the robustness of the network, the designed network is optimized. Furthermore, it is compared with other classical networks to verify the superiority of the network designed in this paper. The structure of this paper is as follows. Section 2 summarizes the basic theory of CNN. In Section 3, the proposed coal-rock recognition method is described in detail. In Section 4, some experiments are conducted to verify the effectiveness of proposed method. The conclusions are summarized in Section 5.

2. Convolutional Neural Network

2.1. Basic CNN Architecture. A convolutional neural network is a kind of trainable feed-forward neural network [22]. A general convolutional neural network consists of an input layer, convolutional layers, pooling layers, fully connected layers, and an output layer. As shown in Figure 1, the first layer of a typical convolutional neural network is the input layer, which is usually followed by a structure that combines multiple convolutional and pooling layers. The last layer is the fully connected layer, and the SoftMax classifier is used to identify the coal and rock. There are two stages in identifying coal and rock using convolutional neural networks: feature learning and classification. The first stage consists of convolutional layer, pooling layer, and fully connected layer, and the last stage contains the SoftMax classifier.

The input layer is used to preprocess the data while inputting the raw data into the neural network. The process of preprocessing is generally the image as a vector and normalizing it in order to increase the speed of the convolutional neural network in training.

The convolutional layer is most important in the convolutional neural network. It is intended to generate a feature map by a convolution operation of a set of weighted filters. Different filters can be used to obtain different feature maps. In principle, the convolution operation is a mathematical operation of point-multiplication summation of two-pixel matrices, where one matrix is the input data matrix and the other matrix is the filter (convolution kernel or feature matrix).

The primary function of the activation function is to make the characteristic map of the output have a nonlinear relationship. There are three kinds of saturated nonlinear functions sigmoid, softsign, and tanh, and unsaturated nonlinear function ReLU [23]. ReLU is commonly used as an activation function in convolutional networks due to the advantage of the speed of unsaturated nonlinear functions when the training gradient descends. The function form of ReLU is shown as equation (1), where \( x \) is the input of the activation function.

\[
f(x) = \begin{cases} 
0, & \text{if } x < 0, \\
 x, & \text{if } x \geq 0.
\end{cases}
\]

As a subsampling process, the pooling layer is used to screen the features in the perception domain and extract the most representative features in the region, which can effectively reduce the output feature scale, thus reducing the number of parameters required by the model and maintaining translation invariance. According to the type of operation, it is usually divided into average pooling and max pooling. Scherer et al. [24] have compared the two merging methods and found that the maximum pooling has faster convergence rate and better generalization ability. Therefore, the maximum pool method is adopted in this paper.

The last layer of the convolutional neural network is the fully connected layer, which is responsible for summarizing the features learned by the convolutional neural network. Finally, the features are classified by using the SoftMax classifier. Assuming that the lengths of the input vectors and output vectors are \( M \) and \( N \), respectively, the number of parameters in the fully connected layer is calculated as follows:

\[
Q = M \times N + N.
\]

In dealing with two-state to polymorphic problems, the extended SoftMax regression algorithm is always the first choice for classification. The training set consists of a label sample \( \{(x^{(1)}, y^{(1)}), \ldots, (x^{(k)}, y^{(k)})\} \), where the value of label \( y \) is 0 or 1 and the input characteristic is \( x^{(i)} \in \mathbb{R}^{n+1} \). We assume that the logistic regression function is as follows:
where $\theta$ is the model parameters of the minimum loss function $J(\theta)$ after training.

$$J(\theta) = \frac{1}{k} \sum_{i=1}^{k} y^{(i)} \log h_{\theta}(x^{(i)}) + \left(1 - y^{(i)}\right) \log \left(1 - h_{\theta}(x^{(i)})\right).$$

(4)

For a multistate classification problem, we assume that there are $n$ conditions and $n$ corresponding labels in the SoftMax classifier. The training set consists of a label sample $\{(x^{(1)}, y^{(1)}), \ldots, (x^{(k)}, y^{(k)})\}$, where the label is $y^{(i)} \in \{1, 2, \ldots, n\}$. For a given training sample $x$ with $n$ classes, the occurrence probability of state $i$ is $P(y = i | x)$, and the output of SoftMax regression is as follows:

$$h_{\theta}(x) = \begin{bmatrix} P(y = 1 | x; \theta) \\ P(y = 2 | x; \theta) \\ \vdots \\ P(y = n | x; \theta) \end{bmatrix} = \begin{bmatrix} e^{\theta_1} \\ e^{\theta_2} \\ \vdots \\ e^{\theta_n} \end{bmatrix} \frac{1}{\sum_{j=1}^{n} e^{\theta_j}},$$

(5)

where $\theta_1, \theta_2, \ldots, \theta_n$ are the model parameters, $(1/\sum_{j=1}^{n} e^{\theta_j})$ playing the role of normalization. Thus, the loss function $J(\theta)$ is as follows:

$$J'(\theta) = \frac{1}{k} \sum_{i=1}^{k} \sum_{j=1}^{n} y^{(i)} = j \log \frac{e^{\theta_j}}{\sum_{j=1}^{n} e^{\theta_j}},$$

(6)

where $\theta_1, \theta_2, \ldots, \theta_n$ are the model parameters of the minimum loss function $J(\theta)$ after training to achieve SoftMax classification.

Since the 1980s, it has been proven that neural networks with multiple hidden layers could be trained by the backpropagation of stochastic gradient descent (SGD) [25]. Each unit in a neural network consists of the relative smooth functions of the input and internal weights. The gradient calculation of the loss function with respect to the weights of multilayer networks can be calculated by backpropagation of derivative chain rule layer by layer. The gradient can only estimate a few examples at a time (not the entire training set), so SGD is becoming one of the most widely used gradient methods in practice.

Figure 2 shows the training process of a convolutional neural network. The input signal is forward-propagated in the neural network and the output data can be obtained through multiple convolutional neural network layers. By comparing the obtained output data with the expected label, the generated error is transmitted layer by layer through backpropagation. The corresponding weights can be updated and the error decreases with the increase of the number of iterations. Finally, the training of the convolutional neural network ends in convergence.

For the $L$-layer of the convolutional neural network, the update formula for the weight between input and output can be expressed as

$$\Delta w_{ij} = \alpha \delta_j X_i,$$

(7)

If the $L$-layer is the last layer of the convolutional neural network, it is calculated as

$$\delta_j = (T_j - Y_j)J'(X_i),$$

(8)

where $T_j$ is the desired label and $J'(X_i)$ is the reciprocal of the activation function. If the $L$-layer is not the last layer of the convolutional neural network, then it is calculated as...
2.2. Network Optimization Strategy. In the application of CNN, the problem of overfitting often occurs. In order to solve this problem, three regularization methods are introduced in this paper: dropout, weight regularization, and batch normalization [26–28].

2.2.1. Dropout. The dropout method makes some neurons temporarily not output with a certain probability \( p \) and then conducts learning and training in the remaining neurons with a probability of \( 1 - p \). After that, all the neurons are restored, and when the next training is performed, some neurons are also randomly selected with probability \( p \) to temporarily not output, and the process is repeated. Using the dropout method is equivalent to selecting a different network structure for each training, which reduces the adaptability and dependence of neurons on each other and enhances the robustness of network model.

2.2.2. Weight Regularization. According to Occam’s razor principle, it can be concluded that the simple model is less likely to overfit than the complex model. Therefore, a common way to reduce overfitting is to force the model weights to take only smaller values, thus limiting the complexity of the model, which makes the distribution of weight values more regular. This method is called the weight regularization and is implemented by adding costs associated with larger weight values to the network loss function. The commonly used regularization method is L2 regularization.

In the L2 regularization method, the added cost is proportional to the square of the weight coefficient. The mathematical expression is as follows:

\[
J(\omega; X, y) = L(\omega; X, y) + \frac{\alpha}{2} w^T w,
\]

where \( L(\omega; X, y) \) represents the loss function, \( (\alpha/2)w^T w \) represents the L2 regularization, and \( w = (w_1, w_2, \ldots, w_n) \) are the parameters of the model. The coefficient \( \alpha \) is set as 0.005.

2.2.3. Batch Normalization. Batch normalization is performed in the normalized layer, which is a learnable network layer with parameters \( \gamma \) and \( \beta \). It is used to normalize the output characteristics of the previous layer into the data with mean 0 and variance 1 and then input to the next layer of the network. The calculation process is as follows:

\[
\begin{align*}
\mu &= \frac{1}{m} \sum_{i=1}^{m} x_i, \\
\sigma^2 &= \frac{1}{m} \sum_{i=1}^{m} (x_i - \mu)^2, \\
\tilde{x}_i &= \frac{x_i - \mu}{\sqrt{\sigma^2} + \epsilon}, \\
y_i &= \gamma \tilde{x}_i + \beta,
\end{align*}
\]

Adding the batch standardization after the convolutional layer has many advantages, which can be concluded as follows:

(i) The convergence speed is becoming faster, so a larger initial learning rate can be selected to increase the training speed.

(ii) It can reduce the network’s dependence on parameter initialization.

(iii) As a form of regularization, batch standardization can reduce the need for dropout in solving the selection problem of regular term parameters in overfitting and improve the generalization ability of the network.

In order to increase the performance of the network model, Szegedy et al. and the colleagues at Google developed an architecture type for convolutional neural network, named Inception, inspired from the early network-in-network architecture [29]. The most basic form of the Inception module consists of 3 to 4 branches, first a 1 × 1 convolution and then a 3 × 3 convolution, and finally the resulting features are joined together. This setup helps the network to learn spatial features and channel-by-channel features.
separately, which is superior to a single convolutional layer when extracting features.

3. Methodology

Through foregoing method, a coal-rock discrimination method is proposed based on a deep convolutional neural network and the flowchart is shown in Figure 3. The method can be divided into four steps: (1) data acquisition, (2) data processing, (3) construction and training of CNN model, and (4) discrimination between coal and rocks.

3.1. Data Acquisition and Processing. Due to the harsh underground environment, it is challenging to acquire enough coal-rock images from the actual coal mining face. In this paper, some artificial coal-rock specimens are produced to construct the coal-rock image samples. In the training of
CNN, if the coal-rock image samples are too few, the overfitting phenomenon will inevitably occur, and the robustness is extremely poor. If the training set contains infinite sample data, the network can observe all contents of the data distribution, so it is difficult to overfit. In order to avoid overfitting and acquire better robustness of the network, it is necessary to enrich the coal-rock image information by means of data augmentation. Data augmentation can generate more training data from existing training samples by using a variety of random transformations that can generate credible images. The goal is that the network will not view the exact same image twice in training so as to make the network have better robustness. In this paper, the data augmentation methods used mainly include adding noise, image scaling, and image rotation.

3.2. Construction and Training of CNN Model. The training of convolutional neural network is the key to achieve the discrimination between coal and rocks. In the training process of CNN, there are two stages: forward learning and backpropagation. In the forward learning stage, a convolutional neural network with N convolution and pooling layers is constructed and the corresponding parameters are initialized. In the second stage, the weight is updated by using the backpropagation from the previous layer, and the error gradient relative to the output node of the current layer is calculated. The error gradient is then passed back to the output node of upper layer using the chain derivation rule. In order to avoid overfitting and speed up the convergence, the regularization operation is adopted to optimize the network. At the same time, the Inception module is also added to the network to improve the classification accuracy and enable the network to extract more advanced abstract features.

3.3. Discrimination between Coal and Rocks. After training the constructed convolutional neural network, the prepared coal-rock image set is used to test the classification performance so as to achieve the intelligent coal-rock recognition. In order to evaluate the performance of the network for coal-rock discrimination, the precision, recall, and F-measure will be used in this paper.

4. Experimental Validation

4.1. Coal-Rock Images Collection. In order to collect the more real sample data, a shearer cutting coal-rock experiment system was established as shown in Figure 4. According to the underground coal mining system, the experimental device mainly includes shearer, hydraulic support, scraper conveyor, artificial coal-rock specimens, and specimen fixing device.

Due to the poor conditions of coal mining working face, it is difficult to directly collect and transport natural coal and rock with large-scale structure and regular form. Therefore, in this experiment, artificial coal-rock specimens were poured with different ratios of coal and cement according to the similarity criterion. The size of each specimen was 1000 mm × 700 mm × 700 mm. In the experiment, the coal-rock specimens were cut in the experimental system, and the images of coal-rock surface were shown in Figure 5.

4.2. Coal-Rock Image Processing. The main task of this paper is to learn different characteristics of coal and rock images by using CNN and then judge that the current coal seam is coal, rock, or coal-rock mixture. Therefore, it is necessary to segment the collected coal-rock interface images. There are 300 original images with a size of 4032 × 3024. There are 6000 images after segmentation, and their size is 256 × 256. Considering the irregular shape of the junction between coal and rock in the specimens during segmentation, the segmented images are divided into three categories: coal image, coal-rock mixed image, and rock image. The ratio is 1 : 1 : 1, as shown in Figure 6. The data set is then divided into 4 parts for the purpose of cross-contrast verification experiments. In the experiment, one set is selected as the testing set at a time, and the other set is used as the training set. The training set is used to implement the training and parameter selection of coal-rock recognition network, and the testing set is used to evaluate the classification effect.

However, as a CNN requires a large amount of training samples, the images acquired from shearer cutting coal-rock specimens cannot meet the algorithm requirements. In order to reflect the coal seam characteristics of coal mining face more truly, some data augmentation methods are adopted for the collected images, which can make CNN easier to learn a variety of features during training. The image processing methods used in this paper include noise adding, image scaling, and image rotation, as shown in Figure 7.

Adding Gaussian noise to an image means adding a Gaussian-like noise to the grey value of a pixel in the image. Typical noise models mainly include Gaussian noise, Poisson noise, and salt and pepper noise. So, this paper added these three noises to the original image.

Scaling an image means that the pixel point in the image is centered at a certain point and is horizontally scaled by times and vertically by times. After scaling, the horizontal distance of the coordinate from the center point becomes times of the original distance, and the vertical distance becomes times of the original distance. In this article, the image is scaled at the origin.
Rotating an arbitrary angle refers to the process of rotating an image around a point to form a new image. The RGB of the pixels of the image before and after the rotation does not change. In this article, the image is rotated around the origin.

4.3. Convolutional Neural Network Architecture. The structural parameters of proposed CNN are listed in Table 1 and we refer to this convolutional neural network as NET, which includes one input layer, five convolutional layers, five maximum pooling layers, three inception modules, one fully connected layer, and one output layer. The model structure of NET can be divided into three parts: the detailed feature extraction of convolution pooling group, the high-level abstract feature extraction of Inception module group, and the output portion of the mapping from feature to classification.

In the first part, the data of the input layer are the 256×256 RGB images, and then they pass through two convolutional layers. In the convolutional layer, 64 convolution kernels with size of 3×3 are used to extract the features of the input images, and similar filling is applied at the same time. ReLU function is selected as the activation function. After extracting features from two convolutional layers, the output feature map size is 256×256 and the dimension is 64. The feature is then compressed as a map size of 128×128 by using a maximum pooling layer with a size of 2×2 window, and the dimensions remain unchanged. After that, it passes through two convolutional
layers again. The number of convolution kernels is 128 and the other parameters are similar to foregoing statement. Finally, the output feature map size is $64 \times 64$ and the dimension is 128.

After the previous convolution and pooling operations, the Inception module is executed, which has 4 branches. The first branch has one convolutional layer consisting of 64 convolution kernels with size $1 \times 1$. The second branch has

Figure 7: Data augmented image. (a) The original image. (b) The Gaussian noise added image. (c) The Poisson noise added image. (d) The salt and pepper noise added image. (e) Image scaling. (f) Image rotation.
two convolutional layers, and the two convolutional layers consist of 92 convolution kernels with size $1 \times 1$ and 128 convolution kernels with size $3 \times 3$. The third branch has two convolutional layers, and the two convolutional layers are composed of 16 convolution kernels with size $1 \times 1$ and 32 convolution kernels with size $5 \times 5$. The fourth branch includes a max pooling and convolutional layer, wherein the convolutional layer consists of 32 convolution kernels with size $1 \times 1$. At this time, the size of the output feature map has changed to $64 \times 64$ and the dimension is 256. The feature map will be compressed with max pooling and the above operations will be repeated twice. The final output map size is $8 \times 8$ and the dimension is 256.

In the last part, the feature graph firstly passes through the convolutional layer composed of 512 convolution kernels with size $1 \times 1$ to achieve higher dimensions. Then Flatten is used to make the feature one-dimensional so that it can better connect with the fully connected layer. It is followed by a fully connected layer with size 256 to distribute the abstract features. Finally, a SoftMax classifier is carried out to output the results.

The parameters of the training network are shown in Table 1. After selecting the training set for data augmentation, the database can contain 31,500 images. In order to optimize the network, in the process of convolution, L2 regularization is added, and the batch normalization operation is performed on the data after the maximum pooling. Finally, the dropout layer is added in front of the fully connected layer to suppress the overfitting problem. When the dropout value is set to 0.5, the network structure generated randomly is the most appropriate. Therefore, the dropout value in this paper is 0.5. For ease of notation, we call this presented network “NET” in the following, and the unoptimized counterpart is denoted as “original NET.”

The learning rate is an important hyperparameter for network training. Usually, appropriate learning rates can speed up network training and achieve better accuracy. Too large or too small learning rate will directly affect the convergence of the network. When the network is trained to a certain stage, the loss function will not be reduced. At this time, the network may encounter two situations. In the first case, the loss function reaches the local optimum. If the local optimum is close to the global optimum, the network can obtain excellent performance, but if the gap is large, the network performance needs to be improved. In the second case, the loss function falls into the saddle point, and the performance of the network is poor. If there is still a fixed learning rate, the model will not be able to continue optimization. To better train the network, we need to change the learning rate during the training. The attenuation method chosen in this paper is expressed as follows:

$$\alpha_{E+1} = \alpha_1 \times e^{-\left((1+\varepsilon)/D\right)}$$

(12)

In the formula, $\alpha_1$ represents the initial learning rate, and the learning rate decays $F$ every $D$ iterations, and $\alpha_{E+1}$ is the learning rate at the $E$-th iteration. Among them, according to the SGD, the initial learning rate $\alpha_1$ is generally chosen to be 0.01. Therefore, the real time learning rate mainly depends on parameters $F$ and $D$. In the simulation, in order to explore the influence of these two parameters on the network, the setting parameters are as shown in Table 2, and five sets of experiments were performed. The accuracy and loss function value during the training process are shown in Figure 8.

It can be seen from Figure 8 that the network exhibits the best performance with higher accuracy and smaller loss function value, when the corresponding parameters are set to 0.5 and 20. The comparison also shows that the performance of simulation c with parameters of 0.2 and 20 is the

| Layer types | Output size | Patch size | Learning parameters |
|-------------|-------------|------------|---------------------|
| Input       | (256 × 256 × 3) | —          | Initial learning rate = 0.01 |
| Conv1       | (256 × 256 × 64) | (64, 3, 3) | Learning rate decay in each of 20 iterations = 50% |
| Conv2       | (256 × 256 × 64) | (64, 3, 3) | |
| MaxPool1    | (128 × 128 × 64) | (2, 2)     | |
| Conv3       | (128 × 128 × 128) | (128, 3, 3) | |
| Conv4       | (128 × 128 × 128) | (128, 3, 3) | |
| MaxPool2    | (64 × 64 × 128) | (2, 2)     | |
| Inception1  | (64 × 64 × 256) | —          | |
| MaxPool3    | (32 × 32 × 256) | (2, 2)     | |
| Inception2  | (32 × 32 × 256) | (512, 3, 3) | |
| MaxPool4    | (16 × 16 × 256) | (2, 2)     | |
| Inception3  | (16 × 16 × 256) | (512, 3, 3) | |
| MaxPool5    | (8 × 8 × 256) | (2, 2)     | |
| Conv5       | (8 × 8 × 512) | (512, 1, 1) | |
| Flatten     | (512 × 1 × 1) | —          | |
| Dense       | (256 × 1 × 1) | —          | |
| Output      | 3           | —          | |

| Experiments | $F$-value | $D$-value |
|-------------|-----------|-----------|
| a           | 0.5       | 20        |
| b           | 0.8       | 20        |
| c           | 0.2       | 20        |
| d           | 0.5       | 30        |
| e           | 0.5       | 10        |
worst, and the reason may be that the learning rate is too large and falls into the saddle point. Simulation c shows that the convergence speed of the network is slower, which is caused by the excessive drop in the learning rate. The performance of simulations d and e is not much different, but there is a significant gap compared to the optimal experiment a, which may be due to the local optimum. Therefore, experimental parameters in this paper are selected as the parameters in simulation a.

4.4. Discussion. In this simulation, three comparative experiments are conducted and the experimental setups are listed in Table 3. Experiment 1 is to explore the impact of the original data on the network performance after enhancement. The raw data (without data augmentation) and augmented data are used to train the CNN with NET, respectively. In Experiment 2, the enhanced data are fed into the NET and the CNN without optimization strategy (original NET) to verify the feasibility of network optimization method for improving network performance. Experiment 3 is to verify the superiority of the proposed network model in terms of the recognition accuracy of coal-rock images. The K-fold cross validation with $K=3$ is used in the experiments and three cross validations ($K=1, 2, 3$) are conducted in the following experiments.

In Experiment 1, the structure and parameters of the network are set as Table 1, and two simulations are performed. The first is that the training set is augmented. After the convergence of network training, the network is tested with the testing set and the result is shown in Table 4. The second is that the training set has not been augmented. After the convergence of network training, the network is tested with the testing set. The result is shown in Table 5.

From Tables 4 and 5, we can find that models trained with augmented data have significantly better performance than models trained with raw data. The reason for this is that there are too few samples of the original data set, and overfitting may occur in the process of training the network, so the network does not perform well in the testing phase. It is concluded that the method of image augmentation can improve the accuracy of the convolutional neural network in identifying coal-rock images.
In Experiment 2, Table 6 shows the performance of the model trained without the optimization strategy in the test set.

From Tables 4 and 6. We can find that, after optimizing the network, the precision, recall, and F1-score were, respectively, increased from 73.49%, 84.67%, and 78.62% to 83.08%, 90.67%, and 86.61%. The reason for this is that dropout, L2 regularization, and batch normalization can generalize the network and improve the robustness of the network. It is concluded that using the optimization strategy to train the network can improve the robustness of the convolutional neural network in identifying coal-rock images.

In Experiment 3, in order to verify the superiority of the proposed network model, we used the enhanced data to train three classical network structures of VGG, GoogleNet, and ResNet. After the network training is completed, the testing set is constructed and enhanced by using adding noise, image scaling, and image rotation. Some experiments are provided and the comparisons with other classical convolutional neural networks are conducted. The results show that the network we designed has better performance in identifying the coal-rock images.

### 5. Conclusions

This paper presents a method for identifying coal and rock based on a deep convolutional neural network. In order to solve the overfitting problem of CNN, three regularization methods of dropout, L2 regularization, and batch normalization are used in this paper. Then the coal-rock image data set is constructed and enhanced by using adding noise, image scaling, and image rotation. Some experiments are provided and the comparisons with other classical convolutional neural networks are conducted. The results show that the network we designed has better performance in identifying the coal-rock images.

### Data Availability

The data used to support the findings of this study are included within the article.

### Conflicts of Interest

The authors declare that they have no conflicts of interest.

### Acknowledgments

The authors would like to thank the National Natural Science Foundation of China for their support to the research work. This work was supported by the Fundamental Research Funds for the Central Universities (no. 2017QNA12) and the Priority Academic Program Development (PAPD) of Jiangsu Higher Education Institutions.

### References

[1] L. Si, Z. Wang, C. Tan, and X. Liu, “A novel approach for coal seam terrain prediction through information fusion of improved D-S evidence theory and neural network,” *Measurement*, vol. 54, pp. 140–151, 2014.

[2] J. Xu, Z. Wang, C. Tan et al., “Cutting pattern identification for coal mining shearer through sound signals based on a convolutional neural network,” *Symmetry*, vol. 10, no. 12, pp. 736, 2018.

[3] L. Si, Z. Wang, X. Liu, and C. Tan, “A sensing identification method for shearer cutting state based on modified multiscale fuzzy entropy and support vector machine,” *Engineering Applications of Artificial Intelligence*, vol. 78, pp. 86–101, 2019.

[4] J. Yang, P. Ren, D. Zhang et al., “Neural aggregation network for video face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4362–4371, Honolulu, HI, USA, July 2017.

[5] Z. Chen, W. Huang, and Z. Lv, “Towards a face recognition method based on uncorrelated discriminant sparse preserving projection,” *Multimedia Tools and Applications*, vol. 76, no. 17, pp. 17669–17683, 2017.
[6] Z. Xu, C. Hu, and L. Mei, "Video structured description technology based intelligence analysis of surveillance videos for public security applications," *Multimedia Tools and Applications*, vol. 75, no. 19, pp. 12155–12172, 2016.

[7] Y. Miao, X. Tao, X. Xu et al., “Joint 3-D shape estimation and landmark localization from monocular cameras of intelligent vehicles,” *IEEE Internet Things*, vol. 6, no. 1, pp. 15–25, 2018.

[8] M. T. N. Truong and S. Kim, "Automatic image thresholding using Otsu's method and entropy weighting scheme for surface defect detection," *Soft Computing*, vol. 22, no. 13, pp. 4197–4203, 2018.

[9] D. S. Kermany, M. Goldbaum, W. Cai et al., "Identifying medical diagnoses and treatable diseases by image-based deep learning," *Cell*, vol. 172, no. 5, pp. 1122–1131e9, 2018.

[10] C. Bartz, H. Yang, and C. Meinel, “SEE: towards semi-supervised end-to-end scene text recognition,” in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, New Orleans, LA, USA, February 2018.

[11] L. H. Dong and P. B. Zhao, “Application of improved Canny edge detection algorithm in coal-rock interface recognition,” *Applied Mechanics and Materials*, vol. 220–223, pp. 1279–1283, 2012.

[12] G. H. Xue, B. H. Hu, X. Y. Zhao, E. M. Liu, and W. J. Ding, "Study on characteristic extraction of coal and rock at mechanized top coal caving face based on image gray scale," *Applied Mechanics and Materials*, vol. 678, pp. 193–196, 2014.

[13] H. Qingyu, Z. Yuanzhong, and C. Jian, "Coal-rock interface identification based on image multi-wavelet transformation," *Industry and Mine Automation*, vol. 41, no. 2, pp. 50–53, 2015.

[14] J. P. Sun and Y. Yang, "A coal-rock image feature extraction and recognition method," *Industry and Mine Automation*, vol. 25, no. 1, pp. 263–273, 2017.

[15] Y. Wu and H. Zhang, “Recognition method of coal-rock images based on curvelet transform and compressed sensing,” *Journal of China Coal Society*, vol. 42, no. 5, p. 33, 2017.

[16] Q. V. Le, M. A. Ranzato, R. Monga et al., "Building high-level features using large scale unsupervised learning," 2011, https://arxiv.org/abs/1112.6209.

[17] M. Abadi, P. Barham, J. Chen et al., “Tensorflow: a system for large-scale machine learning,” in *Proceedings of the 12th Usenix Symposium on Operating Systems Design and Implementation (OSDI’16)*, pp. 265–283, Savannah, GA, USA, November 2016.

[18] S. Ren, K. He, R. Girshick et al., “Faster r-CNN: towards real-time object detection with region proposal networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, pp. 91–99, 2017.

[19] C. Lakshminarayanan and T. Basil, “Feature extraction and automated classification of heartbeats by machine learning,” vol. 1607, 2016, https://arxiv.org/abs/1607.03822.

[20] K. Simonyan, A. Vedaldi, and A. Zisserman, “Deep inside convolutional networks: visualising image classification models and saliency maps,” vol. 1312, p. 6034, 2013, https://arxiv.org/abs/1312.6034.

[21] K. He, G. Gkioxari, P. Dollár et al., “Mask r-CNN,” pp. 2961–2969, 2017, https://arxiv.org/abs/1703.06870.

[22] T. Wiatowski and H. Bölcskei, "A mathematical theory of deep convolutional neural networks for feature extraction," *IEEE Transactions on Information Theory*, vol. 64, no. 3, pp. 1845–1866, 2017.

[23] S. Ioffe and C. Szegedy, “Batch normalization: accelerating deep network training by reducing internal covariate shift,” vol. 1502, 2015, https://arxiv.org/abs/1502.03167.