

Research Article

Identifying Capsule Defect Based on an Improved Convolutional Neural Network

Junlin Zhou,^{1,2} Jiao He,¹ Guoli Li,^{1,3} and Yongbin Liu^{1,3} 

¹College of Electrical Engineering and Automation, Anhui University, Hefei 230601, China

²College of Environmental Science and Optoelectronic Technology, University of Science and Technology of China, Hefei 230026, China

³National Engineering Laboratory of Energy-Saving Motor and Control Technology, Anhui University, Hefei 230601, China

Correspondence should be addressed to Yongbin Liu; lyb@ustc.edu.cn

Received 24 March 2020; Revised 18 May 2020; Accepted 9 June 2020; Published 11 July 2020

Academic Editor: Dong Wang

Copyright © 2020 Junlin Zhou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Capsules are commonly used as containers for most pharmaceuticals, and capsule quality is closely related to human health. Given the actual demand for capsule production, this study proposes a capsule defect detection and recognition method based on an improved convolutional neural network (CNN) algorithm. The algorithm is used for defect detection and classification in capsule production. Defective and qualified capsule images in the actual production are collected as samples. Then, a deep learning model based on the improved CNN is designed to train and test a capsule image dataset and identify defective capsules. The improved CNN algorithm is based on regularization and the Adam optimizer (RACNN), on which a dropout layer and L2_regularization are added between the full connection and the output layer to solve the overfitting problem. The Adam optimizer is introduced to accelerate model training and improve model convergence. Then, cross entropy is used as a loss function to measure the prediction performance of the model. By comparing the results of RACNN with different parameters, a detection method based on the optimal parameters of the RACNN model is finally selected. Results show a 97.56% recognition accuracy of the proposed method. Hence, this method could be used for the automatic identification and classification of defective capsules.

1. Introduction

In the medical industry, the quality of drugs is important and closely related to human health. Medicine and healthcare products in the capsule form play an important role in people's lives. However, capsule products usually feature various defects due to the limitations of capsule production technology [1–3]. Capsule defects include the presence of air bubbles, capsule shrinkage, deformation, and holes. These defects cause unqualified capsule dosage, affect capsule efficacy, and directly lead to poor air tightness, which affect the efficiency and lifespan of the medicine. Poor appearance defects may not affect the efficacy of drugs, but they harm the image of enterprises in the minds of consumers. At present, equipment that can detect capsule defects is few, and methods mainly depend on human visual or sampling inspection. However, these methods have slow detection speed and low detection accuracy and are vulnerable to individual

subjective factors [4, 5]. Therefore, the automatic identification of defective capsules is needed during the production process. In recent years, numerous online capsule detection technologies based on computer vision have been applied [3, 6, 7]. For example, Qi and Jiang proposed a capsule defect detection method based on the hierarchical support vector machine to extract capsule defects in a real-time capsule defect recognition system [2]. Wang et al. used a back-propagation neural network to identify capsule defects [1]. However, these traditional detection methods cannot meet the growing production requirements and, thus, require efficient and reliable detection technology.

As an important part of artificial intelligence, the convolutional neural network (CNN) has achieved processing and feature extraction in image information of complex dimensions [8]. The CNN structure belongs to a multilayered supervised learning neural network, which is changed by a multilayer perceptron. Given its multilevel structure,

the CNN model is suitable for image processing problems [9]. Since the 1990s, CNN has been widely used in face [10, 11], handwriting [12, 13], audio [14], defect [8, 15–18], and fiber recognition [19]. For example, Basheera and Satya Sai Ram proposed a method for classifying brain tumors using the deep features extracted by the CNN [20]. They used the experimental results to prove that the CNN is an important machine learning technology for medical image segmentation and classification. Zhang et al. proposed a method using a CNN model to recognize faces from near-infrared images [21]. Experimental results showed that the proposed CNN structure features higher recognition rate compared with the traditional binary code, Zernike moment, and Hermite kernel recognition methods. Huang et al. applied a CNN to the automatic recognition of Chinese font characters and designed an inception font network with two additional CNN structural elements [13]. In addition, Soukup and Huber-Mörk applied a CNN to steel surface defect recognition [16]. They experimented with a classical CNN trained using a pure supervisory method, discussed the influence of regularization methods, and achieved a high recognition rate for steel surface defects. Zhang et al. proposed a novel dual-channel CNN framework for an accurate spectral spatial classification of hyperspectral images [22]. One- and two-dimensional CNNs were used to extract hierarchical spectral and space-related features, respectively, and the expected classification results were obtained. Kumar et al. presented a framework that uses deep CNNs to classify multiple defects in sewer closed circuit television inspection images and achieved high classification accuracy [23]. These experimental results proved the effectiveness of CNNs in defect recognition.

Given the methods used in the literature, the CNN architecture features remarkable success and speed in classification compared with other methods. In addition, compared with the general neural network, the CNN exhibits notable advantages [24–26]: (a) shared convolution kernel and the absence of pressure in high-dimensional data processing; (b) omission of manual selection of features when using weights that are trained to obtain features; and (c) low number of network connections and training parameters, and a simple network structure. However, CNNs are rarely used for the identification of defects in capsule detection. Therefore, this paper proposes a capsule defect detection method based on an improved CNN algorithm (RACNN). The dropout layer and L2_regularization are added between the full connection and output layers, and the Adam optimizer is introduced to accelerate model training and improve model convergence. The detection method based on the optimal parameters of the RACNN model is finally selected through optimizing parameters. The proposed method achieves high recognition accuracy in the process of capsule defect detection, which provides a new method for defect capsule classification.

2. CNN Methodology

2.1. CNN. The most basic component of CNNs is the neuron model. Two layers of neurons constitute the perceptron

model, which can achieve the logical operation and weight learning of a complex task. Adding a hidden layer between the input and output layers to form a multilayer functional neuron model can solve multiclassification problems [16, 17]. CNN is an artificial neural network based on deep learning theory. The core part of CNNs is the hidden layer, which is composed of convolution, pooling, and fully connected layers.

The convolution layer extracts the main features of the input image data and contains multiple convolution kernels, which are similar to the feed forward neural network neurons in the convolutional layer [24]. The convolution kernel in the convolutional layer can extract deep information from the data and local features of the image. A brief description of the convolution process is described by the following formula:

$$y = w_i^l * x^{(j)}, \quad (1)$$

where $*$ denotes the dot product of the kernel and the local regions, w_i^l denotes the weights of the i th filter kernel in layer l , and $x^{(j)}$ denotes the j th local region in convolutional layer l .

The activation function is essential after the convolution operation. The activation function adds nonlinear factors to the neural network to solve complex problems. In recent years, ReLU has been widely used as an activation unit of CNNs [27]. Compared with the common activation functions, sigmoid and tanh, ReLU offers advantages, such as a low calculation cost. ReLU sets the output value of certain neurons to zero, thereby resulting in the sparseness of the network and reduction in the interdependence of parameters; as such, the problem of overfitting is alleviated. The ReLU activation function is described by the following formula:

$$f(u^{l(i,j)}) = \max\{0, u^{l(i,j)}\}, \quad (2)$$

where $u^{l(i,j)}$ is the output value of the previous operation.

The pooling layer is used to compress the feature matrix while retaining useful information; it can accelerate the calculation and prevent overfitting [28]. Pooling includes two methods: maximum and average. The CNN designed in this study uses maximum pooling. The maximum pooling transformation is described by the following formula:

$$p^{l(i,j)} = \max\{a^{l(i,n)}\}, \quad (j-1)W + 1 \leq n \leq jW, \quad (3)$$

where $a^{l(i,n)}$ denotes the value of the n th neuron in the i th frame of layer l and W is the width of the pooling region.

In most cases, the full connection layer is located at the end of the network. This layer performs regression classification on the features of the previous layer-by-layer transformation and mapping extraction.

2.2. RACNN for Capsule Identification. On the basis of the basic CNN model, the proposed CNN model is improved from three aspects of preprocessing, optimizer, and model structure. The improved CNN algorithm (RACNN) is constructed with regularization and the Adam optimizer.

The RACNN structural model for capsule recognition is shown in Figure 1. The main steps are as follows:

Step 1: in the image preprocessing stage, the resize function is introduced to process the capsule image.

Step 2: the preprocessed image dataset is inputted for training, and the weight w_i and bias b_i of the network are initialized.

Step 3: the RACNN structure is designed with four convolution-pooling and three fully connected layers. The first layer convolutional feature matrix X^1 is calculated, and the feature matrix X^1 is merged into one column vector as the neuron input to the next layer. The weight w_i and the bias b_i are updated to obtain the feature matrix X^2 , and then the next layer is sequentially executed. The specific parameters of each layer are described in the following chapters.

Step 4: the dropout layer is added between the fully connected and the output layers to improve the normalization ability of the network. In addition to the dropout layer, L2_regularization is introduced to prevent overfitting. The difference is that the dropout is obtained by modifying the neural network itself, whereas L2_regularization is achieved by modifying the loss function. That is, the L2_regularization adds regularization into the original loss function. The principle is shown by the following formula:

$$J = J_0 + C\|W\|_2^2, \quad (4)$$

where J_0 represents the original loss function, C represents the L2_regularization parameter, and W is the weight matrix of each layer in the trained network.

Step 5: in the model training stage, the Adam optimizer is introduced to update and calculate the network parameters, which affect the model output value and make it close to the optimal value. As a result, the model's convergence speed is improved, and the model error is reduced.

Step 6: finally, a SoftMax classifier is used to map the extracted features to 10 different types of capsule images. The output of each layer is fused and input to the SoftMax classifier to improve further the recognition accuracy of capsules.

3. Experimental Design and Parameter Setting

In the production process of the capsule, 10 types of capsule images, including nine types of defective capsules and one type of normal capsule, were acquired. Image enhancement technology was used for image processing to enrich the image training set, extract image features, and promote the model. Two image enhancement techniques, namely, image rotation and image adding noise, were used in the experiment, among which Solt noise and Gaussian noise were added. Figure 2 shows the physical images of the capsules. The types of defective capsules included holes, concave heads, uncut bodies, oil stains, short bodies, insertion,

shrivel, locking, and nesting. Table 1 lists the number of each type of capsule image processed by image enhancement technology for subsequent experiments. Given that the CNN classification of this study belongs to supervised learning, the red number indicates the label set for each capsule category.

The CNN structure used in the experiment comprised four convolution and pooling layers, followed by fully connected hidden and SoftMax layers. The size of the convolution kernel of the first two layers was $5 * 5$, and that of the latter two layers was $3 * 3$. Maximum pooling was used, and the excitation function was ReLU. Table 2 provides the detailed parameters of the convolutional and pooling layers. The experiments were implemented using the TensorFlow toolbox. In the training process of the RACNN model, the k-fold cross validation method was used to evaluate the model, and k was set to 10. All capsule image datasets were randomly divided into 10; nine of them were trained each time, and the remaining one was used for testing (i.e., 1413 and 157 samples were trained and validated, respectively). The process was repeated 10 times, and the image datasets used for testing each time were different. The specific implementation process was completed by calling the k-fold function in the sklern.model_selection module in the Scikit-learn library.

The sample images acquired from the producing spot were input into the RACNN network. Each sample image was preprocessed into a $100 * 100$ pixel image as the input dataset by using the resize function in the Skimage library. The size of the convolution core in the first layer was $5 * 5$, and the convolution layer C1 obtained 32 feature images with $100 * 100$ pixels. The downsampling coefficient was 2, that is, the step length of the pooling layer P2 was 2. The pooling layer P2 obtained 32 feature images with $50 * 50$ pixels. The same process was executed on the next layers. Specific parameter settings are shown in Table 2. As the last layer of RACNN, the SoftMax layer classified the data and outputted a vector of $10 * 1$. Each vector value represented the probability that each sample belonged to a class. Cross entropy was used as the loss function, which increased as the predicted probability diverged from the actual label. Thus, the model aimed to minimize cross entropy. After establishing the RACNN network as described above, training data were used to train the network and fix the trained RACNN network. Finally, the test data were identified and classified by the trained RACNN network.

4. Results and Discussion

4.1. Selection of the Learning Rate and Batch_size. The learning rate and batch_size are important parameters of CNNs. The learning rate considerably affects the test effect of the network. An extremely high or low learning rate reduces recognition accuracy. The learning rate is the step interval of each gradient descent, which determines how far the weight moves in the gradient direction. When the learning rate is large, the convergence speed of the model is faster in the early stage. But, this will make the model difficult to converge, that is, oscillate near the extreme point, and never reach the best point. When the learning rate is small, the

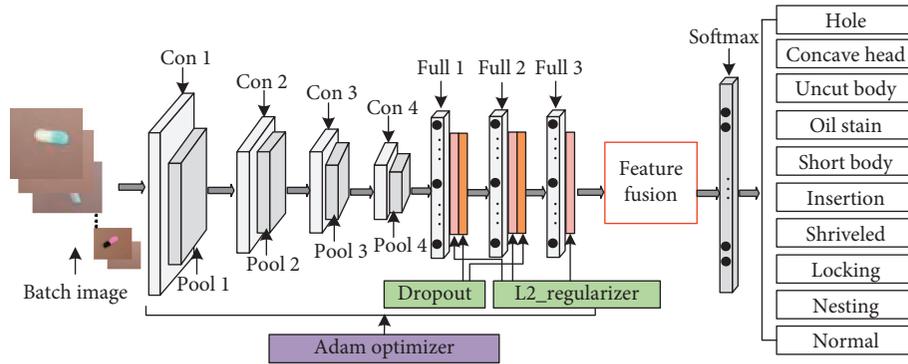


FIGURE 1: RACNN structural model for capsule recognition.

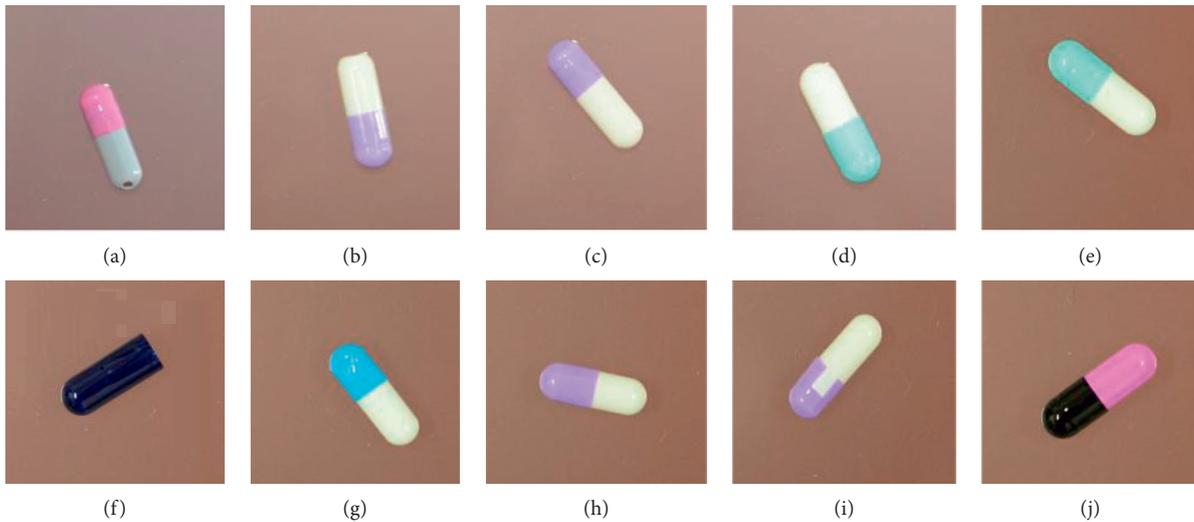


FIGURE 2: Physical images of the defective capsules. (a) Hole, (b) concave head, (c) uncut body, (d) oil stain, (e) short body, (f) insertion, (g) shriveled, (h) locking, (i) nesting, and (j) normal.

convergence rate will be extremely slow. Therefore, in the experiment, a large learning rate is set in the early stage to make the gradient drop rapidly, and then the learning rate is gradually reduced to make the model gradually reach the best. But, at the same time, dropout and regularization were used to prevent overfitting. Different learning rates for training are used to optimize the relationship between the learning rate and recognition accuracy in the experiment. The results are shown in Table 3. As the learning rate decreases, the test accuracy gradually increases, but when the learning rate is too small, the test accuracy decreases. When the learning rate is approximately 0.0007 in experiment, the recognition accuracy is the highest, and the model should be optimal.

Batch_size indicates the batch number of images that the model reads during training or testing. For the classification experiments of different targets, setting the appropriate batch_size plays an important role in improving the training efficiency and recognition performance of the network model. When the batch_size is extremely small, the network convergence is unstable, and the convergence speed is slow. Hence, the loss value fluctuates back and forth. When the batch_size is extremely large, the amount

of calculation is large, memory consumption is excessive, and local optimum may occur. Therefore, in this experiment, after adjustment, the optimal parameters are obtained. The learning rate is 0.0007; the maximum number of training is 20; and the batch_sizes are set to 8, 16, 32, and 64 for comparison. As shown in Table 4, when the batch_size is 16, the smallest loss is obtained, and the test set achieves the highest accuracy.

4.2. Analysis of Results. Figures 3 and 4 show the influence of the learning rate on the experimental results. Figures 3(a) and 3(b) display the accuracy curve of the training and test sets under different learning rates, respectively. When the learning rates are 0.0007 and 0.0004, the accuracy rate is higher under the same number of iterations compared with other learning rates. Moreover, the accuracy improves gradually with the increase in the number of training times. When the number of iterations reaches approximately 11, the recognition accuracy increases slowly and remains stable. Figures 4(a) and 4(b) show the change curve of loss of the training and test sets, respectively, at different learning rates. As shown in the figures, after 20 iterations, the

TABLE 1: Specific quantity of each capsule.

Hole	Concave head	Uncut body	Oil stain	Short body	Insertion	Shriveled	Locking	Nesting	Normal
112	184	182	168	184	154	182	105	193	106

TABLE 2: Details of the proposed CNN model used in the experiments.

Layer	Kernel size	Stride	Padding	Output size	Output number
Convolution 1	5 * 5	1	Yes	100 * 100	32
Pooling 1	2 * 2	2	No	50 * 50	32
Convolution 2	5 * 5	1	Yes	50 * 50	64
Pooling 2	2 * 2	2	No	25 * 25	64
Convolution 3	3 * 3	1	Yes	25 * 25	128
Pooling 3	2 * 2	2	No	12 * 12	128
Convolution 4	3 * 3	1	Yes	12 * 12	128
Pooling 4	2 * 2	2	No	6 * 6	128
Fully connected 1				1 * 1	1024
Fully connected 2				1 * 1	512
Fully connected 3				1 * 1	12

TABLE 3: Effect of learning rate on recognition results.

Learning rate	Train_loss	Train_accuracy (%)	Test_loss	Test accuracy (%)
0.004	8.0523	76.02	16.27	75.98
0.001	0.0026	97.44	2.613	96.14
0.0007	0.0316	100	0.3999	97.56
0.0004	0.0235	100	0.9852	95.22
0.0001	0.5625	99.71	8.5397	89.81
0.00007	0.5955	92.05	9.0707	84.03

TABLE 4: Effect of the batch_size on recognition results.

Batch_size	Train_loss	Train_accuracy (%)	Test_loss	Test accuracy (%)
8	0.0156	100	0.4126	97.02
16	0.0316	100	0.3999	97.56
32	0.0615	100	3.5738	96.09
64	1.9609	100	9.0746	95.31

negligible loss of networks occurs at the learning rates of 0.0007 and 0.0004. With the increase in iterations, the loss value continuously decreases and approaches zero.

Figures 5 and 6 show the influence of the batch_size on the experimental results. Figure 5(a) and 5(b) show the curve of the accuracy of the training and test sets under different batch_sizes, respectively. The recognition accuracy increases slowly and remains stable when the number of iterations reaches approximately 11. Similarly, Figures 6(a) and 6(b) display the curves of the loss of the training and test sets under different batch_sizes, respectively. Figure 6 shows that when the batch_size is large, the initial loss value is also large. Figure 5 shows that when the batch_size is small, the accuracy curve fails to rise smoothly, indicating the instability of the network. Therefore, the network can achieve accurate

and rapid convergence effect only when an appropriate batch_size is selected for our capsule sample set.

4.3. Efficiency Evaluation. As described above, the optimal parameters of the RACNN model in the experiment are as follows: batch_size, 16; learning rate, 0.0007; and maximum training epoch, 20. The recognition accuracy of the test set obtained by the optimal parameter model is 97.56%. The confusion matrix of the test accuracy is drawn after normalization of the capsule sample data to explain the unbalanced classification results of each kind of defective capsule in detail (Figure 7). A specific matrix is used to present the visualization effect of the algorithm performance. The labels of 10 kinds of capsules (order: hole, concave head, uncut body, oil stain, short body, insertion, shriveled appearance, locking, nesting, and normal) are defined as numbers from 1 to 10 in order. The results show that most of the capsules are correctly identified. In the experiment, 14% of the oil-stained capsules is mistaken for uncut body capsules, and 11% of the shriveled capsules is incorrectly identified as hole capsules.

For comparison, RACNN, CNN, support vector machine (SVM), and k-nearest neighbor (KNN) are applied to analyze the same image dataset. The identification accuracy

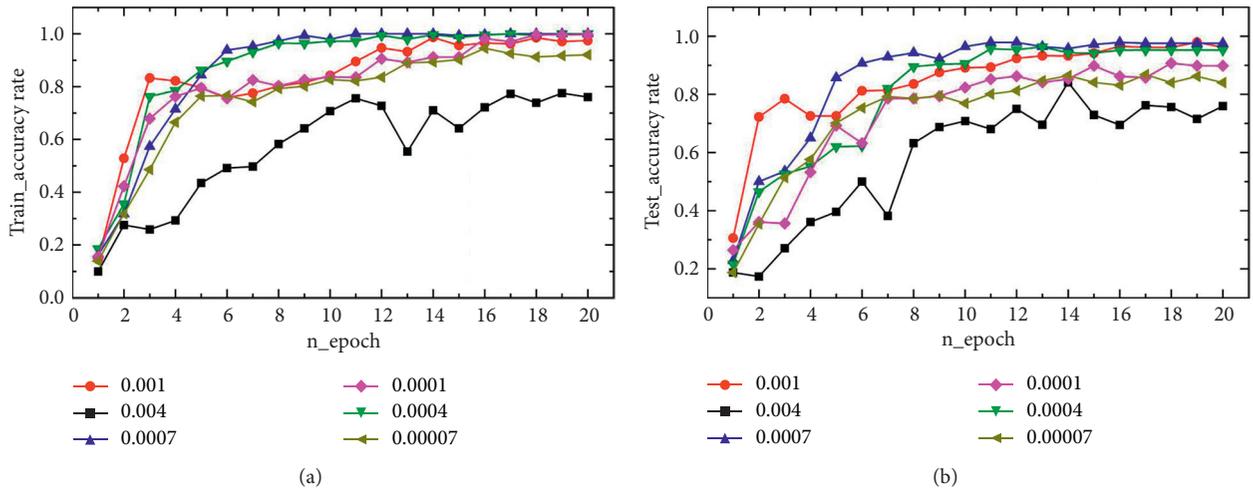


FIGURE 3: Recognition accuracy curve under different learning rates.

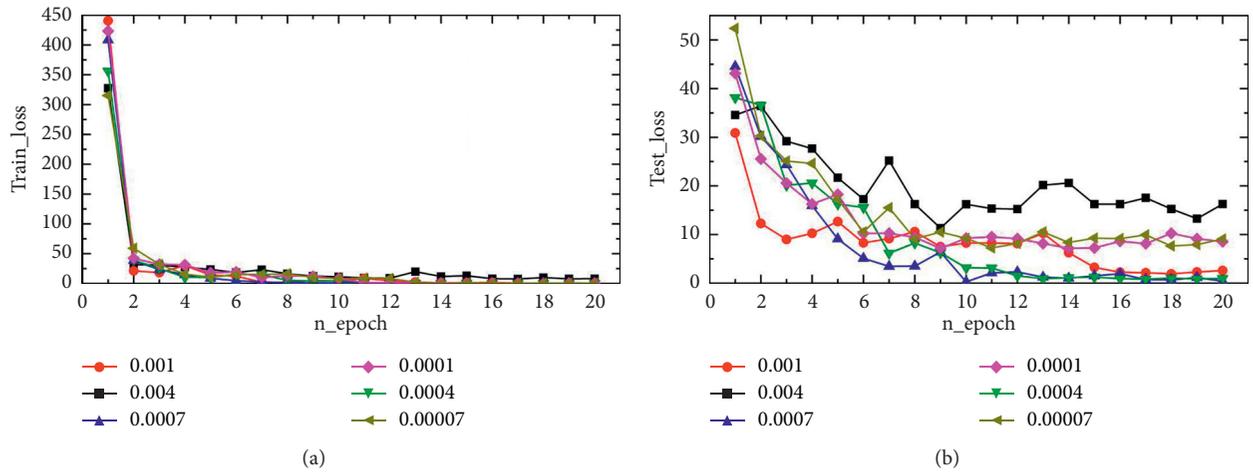


FIGURE 4: Loss curve under different learning rates.

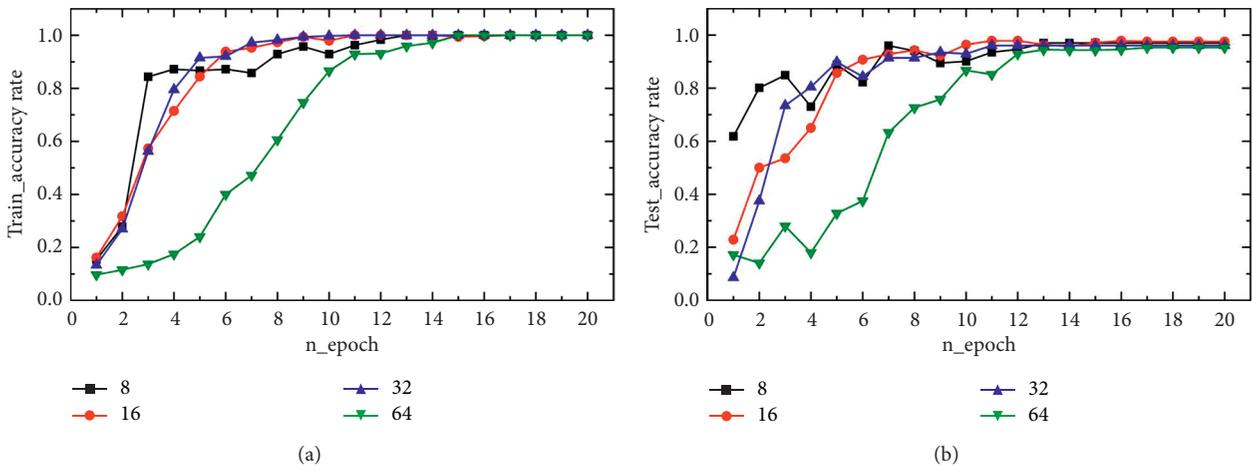


FIGURE 5: Recognition accuracy curve under different batch_sizes.

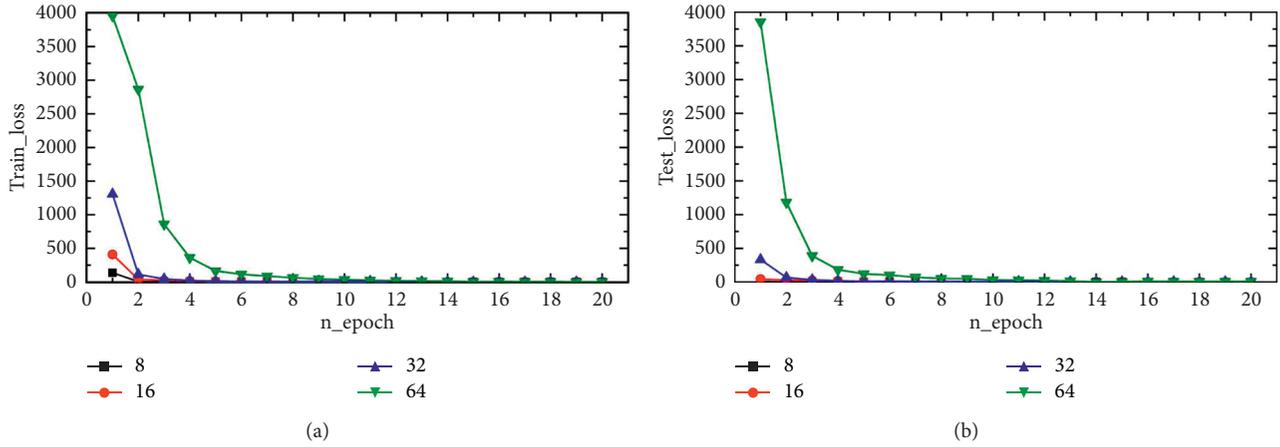


FIGURE 6: Loss curve under different batch_sizes.

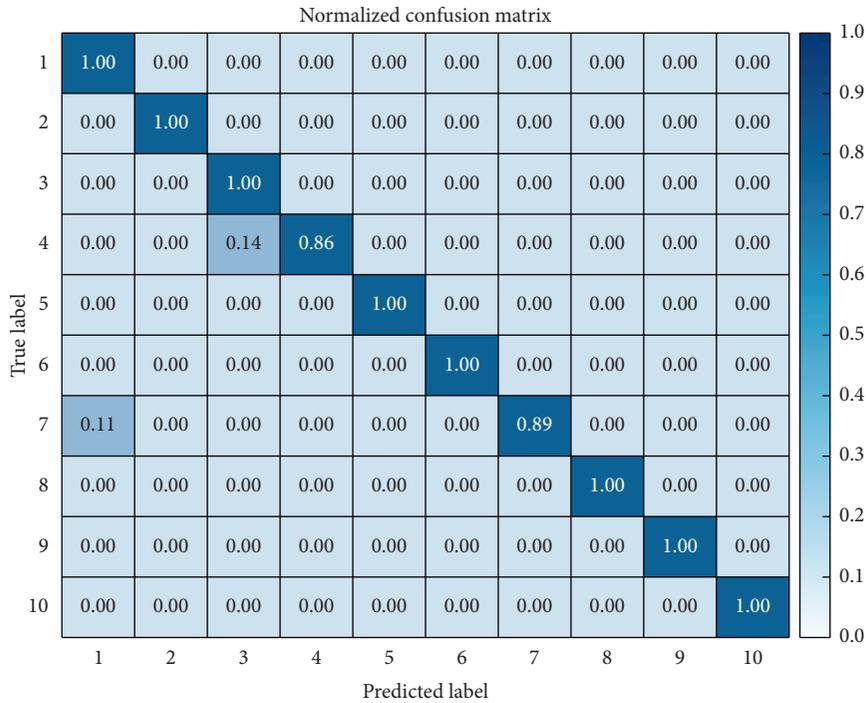


FIGURE 7: Identification results for 10 types of capsules: confusion matrix of CNN.

for the four methods is shown in Figure 8. RACNN obtains the best performance among the four methods. Although the identification accuracy of SVM is relatively high, SVM is not suitable for processing large-scale training samples. Since the number of capsules in actual production is far more than that used in experiments, this method is more complicated when applied to the actual production process

of a capsule. However, compared with SVM, CNN has no pressure on high-dimensional data processing, and is more suitable for processing large-scale sample data. It is concluded from experiments that RACNN has higher identification accuracy, and the classifier is simple and easy to operate, which is convenient for application in actual capsule production.

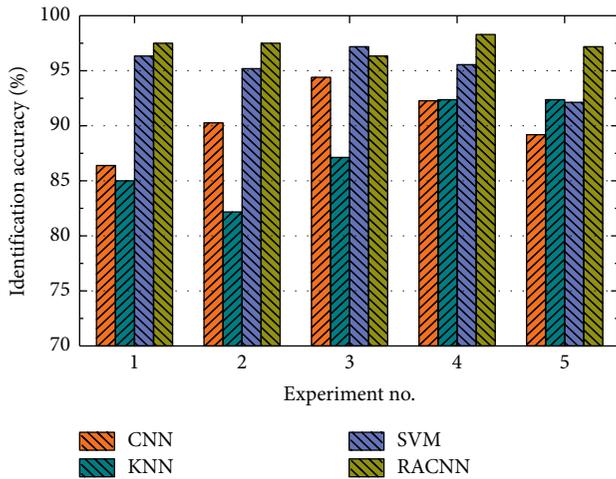


FIGURE 8: Results of different methods.

5. Conclusion

An efficient method for capsule defect detection is crucial to improve capsule quality. Therefore, in this study, a CNN model was introduced to recognize defective capsules. Sample images, including one kind of normal capsule and nine kinds of defective capsules, were collected. An improved CNN algorithm based on regularization and the Adam optimizer was designed. The RACNN model was used to train and identify the capsule images. In addition, the effects of two important network parameters (batch_size and learning rate) on the experimental results were analyzed in detail. By adjusting the parameters, the best parameter combination model was obtained, and the classification accuracy was as high as 97.56%. The experimental results showed that the proposed method is concise and efficient, and the RACNN structure model is suitable for the identification of defective capsules.

Data Availability

Data can be obtained from National Engineering Laboratory of energy saving motor and control technology, College of Electrical Engineering and Automation, Anhui University.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Key Research and Development Plan of Anhui Province (201904A05020034), the National Natural Science Foundation of China (51675001), and the State Key Program of National Natural Science of China (51637001).

References

- [1] H. Wang, X. Liu, and C. Yi, "Detection of capsule foreign matter defect based on BP neural network," in *Proceedings of the IEEE International Conference on Granular Computing*, pp. 325–328, Rome, Italy, October 2014.
- [2] D. Y. Qi and Z. Jiang, "Capsule defects classification based on hierarchical support vector machines," *Advanced Materials Research*, vol. 926–930, pp. 3373–3378, 2014.
- [3] W. Qi, T. Zhang, Z. Cai, J. Nan, and X. Zhang, "Visual quality inspection of capsule heads utilizing shape and gray information," *Journal of Electronic Imaging*, vol. 24, no. 6, Article ID 061121, 2015.
- [4] J. C. Xu, Q. Y. Yue, X. Y. Ren, and Q. Wang, "Design of tablet form merohedral stuffing identification and sorting system based on machine vision," *Transducer and Microsystem Technologies*, vol. 36, 2017, In Chinese.
- [5] D. H. Lai and Y. W. Huang, "Inspection for defected capsules based on extreme learning machine," *Journal of Fuzhou University (Natural Science Edition)*, vol. 40, 2012, In Chinese.
- [6] T. K. Liu, Y. C. Chou, and Y. T. Wen, "Hybrid evolutionary optimization for nutraceutical manufacturing processes," *Journal of Intelligent Manufacturing*, vol. 28, pp. 1–14, 2017.
- [7] X. Gao, J. Mu, Q. Li et al., "Comprehensive identification of guan-xin-shu-tong capsule via a mass defect and fragment filtering approach by high resolution mass spectrometry: in vitro and in vivo study," *Molecules*, vol. 22, no. 6, p. 1007, 2017.
- [8] L. Shang, Q. Yang, J. Wang, S. Li, and W. Lei, "Detection of rail surface defects based on CNN image recognition and classification," in *Proceedings of the 20th International Conference on Advanced Communication Technology*, Chuncheon, South Korea, February 2018.
- [9] T. Parcollet, M. Morchid, and G. Linarès, "Quaternion convolutional neural networks for heterogeneous image processing," in *Proceedings of the IEEE International Conference on Acoustics*, pp. 8514–8518, Calgary, Canada, April 2018.
- [10] Y. Sun, D. Liang, X. Wang, and X. Tang, "DeepID3: face recognition with very deep neural networks," *Computer Science*, vol. 1502, 2015.
- [11] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," *Computer Science*, vol. 1411, p. 7923, 2014.
- [12] L. G. Hafemann, R. Sabourin, and L. E. S. D. Oliveira, "Analyzing features learned for offline signature verification using deep CNNs," in *Proceedings of the International Conference on Pattern Recognition*, pp. 2989–2994, Cancun, Mexico, December 2016.
- [13] S. Huang, Z. Zhong, L. Jin, S. Zhang, and H. Wang, "DropRegion training of inception font network for high-performance Chinese font recognition," *Pattern Recognition*, vol. 77, pp. 395–411, 2018.
- [14] O. Abdel-Hamid, A. R. Mohamed, J. Hui, and G. Penn, "Applying Convolutional Neural Networks concepts to hybrid NN-HMM model for speech recognition," in *Proceedings of the IEEE International Conference on Acoustics*, pp. 4277–4280, Kyoto, Japan, March 2012.
- [15] H. Liu and R. Guo, "Detection and identification of SAWH pipe weld defects based on X-ray image and CNN," *Chinese Journal of Scientific Instrument*, vol. 29, pp. 247–256, 2018.
- [16] D. Soukup and R. Huber-Mörk, "Convolutional neural networks for steel surface defect detection from photometric stereo images," *Advances in Visual Computing*, vol. 8887, pp. 668–677, 2014.
- [17] X. Cui, Y. Liu, Y. Zhang, and C. Wang, "Tire defects classification with multi-contrast convolutional neural networks," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 32, no. 4, p. 1850011, 2018.

- [18] T. Xian, Z. Wang, Z. Zhang, D. Zhang, and Z. Lei, "Wire defect recognition of spring-wire socket using multitask convolutional neural networks," *IEEE Transactions on Components Packaging & Manufacturing Technology*, vol. 8, no. 4, pp. 689–698, 2018.
- [19] X. Wang, C. Zhao, G. Liu, and W. Yan, "Fiber image classification using convolutional neural networks," in *Proceedings of the International Conference on Systems & Informatics*, pp. 1214–1218, Cairo, Egypt, September 2017.
- [20] S. Basheera and M. Satya Sai Ram, "Classification of brain tumors using deep features extracted using CNN," *Journal of Physics Conference Series*, vol. 1172, no. 1, Article ID 012016, 2019.
- [21] X. Zhang, P. Min, and C. Tong, "Face recognition from near-infrared images with convolutional neural network," in *Proceedings of the International Conference on Wireless Communications & Signal Processing*, pp. 1–5, Yangzhou, China, October 2016.
- [22] H. Zhang, Y. Li, Y. Zhang, and Q. Shen, "Spectral-spatial classification of hyperspectral imagery using a dual-channel convolutional neural network," *Remote Sensing Letters*, vol. 8, no. 5, pp. 438–447, 2017.
- [23] S. S. Kumar, D. M. Abraham, M. R. Jahanshahi, T. Iseley, and J. Starr, "Automated defect classification in sewer closed circuit television inspections using deep convolutional neural networks," *Automation in Construction*, vol. 91, pp. 273–283, 2018.
- [24] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proceedings of the International Conference on Neural Information Processing Systems*, pp. 1097–1105, Nevada, CA, USA, December 2012.
- [25] C. Shen, Y. Qi, J. Wang, G. Cai, and Z. Zhu, "An automatic and robust features learning method for rotating machinery fault diagnosis based on contractive autoencoder," *Engineering Applications of Artificial Intelligence*, vol. 76, pp. 170–184, 2018.
- [26] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision & Pattern Recognition*, pp. 4353–4361, Boston, MA, USA, June 2015.
- [27] J. S. Choi and M. Kim, "A deep convolutional neural network with selection units for super-resolution," in *Proceedings of the Computer Vision & Pattern Recognition Workshops*, pp. 154–160, Honolulu, HI, USA, July 2017.
- [28] J. Gu, Z. Wang, J. Kuen et al., "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018.