ISSN 2616-5775 Vol. 6, Issue 2: 16-22, DOI: 10.25236/AJCIS.2023.060203

Rice Leaf Nitrogen Deficiency Image Classification Model Based on CNN

Luo Yuchen[1,a,*], Zan Xusen[2,b], Li Jiali[3,c], Zhang Xueting[4,d]

[1]*College of Mathematical Statistics, Ningxia University, Yinchuan, China*
[2]*College of Physics and Electronic and Electrical Engineering, Ningxia University, Yinchuan, China*
[3] *College of Advanced Interdisciplinary Studies, Ningxia University, Zhongwei, China*
[4]*College of Education, Ningxia University, Yinchuan, China*
[a]*luo.yuchen@qq.com,* [b]*646935835@qq.com,* [c]*1368821170@qq.com,* [d]*zhangxueting2003@163.com*
**Corresponding author*

Abstract: *This study endeavors to utilize a convolutional neural network (CNN) to categorize pictures of nitrogen insufficiency in rice foliage. Thousands of real photos of rice leaves falling under four different labels of nitrogen deficiency served as the basis for this classification model. This study compares the effectiveness of several optimizers on this dataset and provides a detailed description of how a convolutional neural network model is constructed. The accuracy and loss of the model are computed to quantitatively assess its performance. Finally, when the optimizer is switched from Adam to Nadam, the accuracy of the model increases from 94.75% to 99.5%.*

Keywords: *Machine Learning, CNN, Data Processing, Image Classification, Optimizer Comparison*

1. Introduction

Half of the world's population takes rice as their daily diet. According to World Bank projections, demand for rice consumption will increase by 51 percent by 2025. Rice yields are affected by several factors, including mineral deficiencies. Minerals required to maintain standard healthy rice crops are Nitrogen (N), Potassium (K), Phosphorus (P), Boron (B), Zinc (Zn), Sodium (S), Copper (Cu) and Magnesium (Mg) [1]. Nitrogen is one of the mineral elements necessary for the growth and development of rice. To improve productivity, nitrogen fertilizer (N) application in farmland has been increasing for the past 60 years [2]. Sometimes farmers apply excess N in the expectation of greater yields, but this causes stem rot. Therefore, the predicted nitrogen requirements in the fields are important for the normal growth and development of rice. Agronomists mainly use information about leaves to determine plant nutritional status, and leaf sheaths also show specific symptoms when rice is deficient in nutrients [3].

The Leaf Color Chart (LCC) was developed by the International Rice Research Institute (IRRI) to determine nitrogen (N) fertilizer requirements for rice crops. The LCC has five green exchanges, ranging in color from chartreuse to dark green. It determines the green color of rice leaves, which indicates their nitrogen content [4]. Images were collected in daylight using a 13-megapixel smartphone camera following the LCC instructions. The LCC, developed by ICAR-NRRI in Kathakalpa, India, has five interchanges. Swap-5 indicates nitrogen abundance and Swap-1 indicates high nitrogen deficiency. In this study, we consider four swaps from Swap-4 to Swap-1.

In recent years, incredible progress has been made in crop disease diagnosis, such as detection, identification and quantification of various diseases, with the application of computer vision and machine learning techniques. Many researchers have reported methods for the automatic diagnosis of rice diseases based on digital image processing technology [5], SVM [6], and computer vision [7]. The study is not only aimed at classifying rice diseases, but also at other crops such as wheat, corn and tomatoes. Although machine learning technology has made great achievements in image recognition, there are still certain limitations: the data processing capacity is limited, and it needs to be segmented and feature extracted [8]. With the advancement of machine learning techniques, deep learning methods are well-equipped to solve and model big data problems. Deep learning methods can be applied to agricultural disease image classification without pre-requiring processes such as segmentation and feature extraction.

Over the past few years, CNNs have been applied in various fields such as object detection, image classification, and video classification. In addition, some researchers have recently reported advanced

ISSN 2616-5775 Vol. 6, Issue 2: 16-22, DOI: 10.25236/AJCIS.2023.060203

filtering techniques [8], hybrid feature selection methods [9], and powerful classification models such as Voronoi diagram-based classifiers (VDBC) and neuro-fuzzy (NF) classifiers [10].

Our CNN-based model for rice nitrogen deficiency image classification aims to provide a convenient system for farmers. The model is optimized by adapting the optimizer from Adam to Nadam for better image classification, and the performance of the classification model is measured in terms of accuracy and loss rate.

2. Methodology

2.1. Data Preprocessing

Before model training, we first carry out data processing. There were 5,390 images in the training dataset and 400 photos in the test dataset. The quantity distribution of all kinds of pictures is shown in Table 1:

Table 1: Number of Image Samples According to Swap of LCC

Sample classification	Training sample number	Testing sample number
Swap1	1407	100
Swap2	1203	100
Swap3	1400	100
Swap4	1380	100
Total	5390	400

To make the images captured by the model more comprehensive and accurate, we adjusted all the images to a resolution of 150×150. During the training, 5 images were randomly selected from different classes in the training data set and iterated successively for image processing.

2.2. Construction of Convolutional Neural Networks

2.2.1. Convolution Layer

Keras library initializes the weight w and bias b of the convolutional kernel by itself when we construct the convolutional layer. The neurons between the convolutional layers are locally connected and share weights. In our model, there are four convolutional layers and four pooling layers respectively.

Parameters of the convolutional layer are set as follows:

· filter number(K): The initial number of filters in the convolution is 32, and the number of filters gradually increases with the increase of the number of layers, which gradually enhances the intensity of model training.

· filter size(F): The size of the convolution kernel is 3×3.

· Strider(S): strider defaults to 1.

· Zero pad(P): zero pad defaults to 0.

· The activation function is Relu, and its function expression is:

$$f(x) = max(0, x) \qquad (1)$$

· The size of the input image($a_1 \times b_1 \times c_1$): 150×150×3.

The convolution kernel K is convolved with a part of the input matrix, and then the result is mapped to the convolution layer to obtain the feature map after convolution. Given the above parameters of the convolution layer, each index $a_2 \times b_2 \times c_2$ of the output feature graph can be calculated as 148×148×32 according to the following formula:

$$a_2 = \frac{a_1 - F + 2P}{S} + 1 \qquad (2)$$

$$b_2 = \frac{b_1 - F + 2P}{S} + 1 \qquad (3)$$

$$c_2 = K \qquad (4)$$

ISSN 2616-5775 Vol. 6, Issue 2: 16-22, DOI: 10.25236/AJCIS.2023.060203

Meanwhile, the process of convolution operation in the convolution kernel is shown in Figure 1:

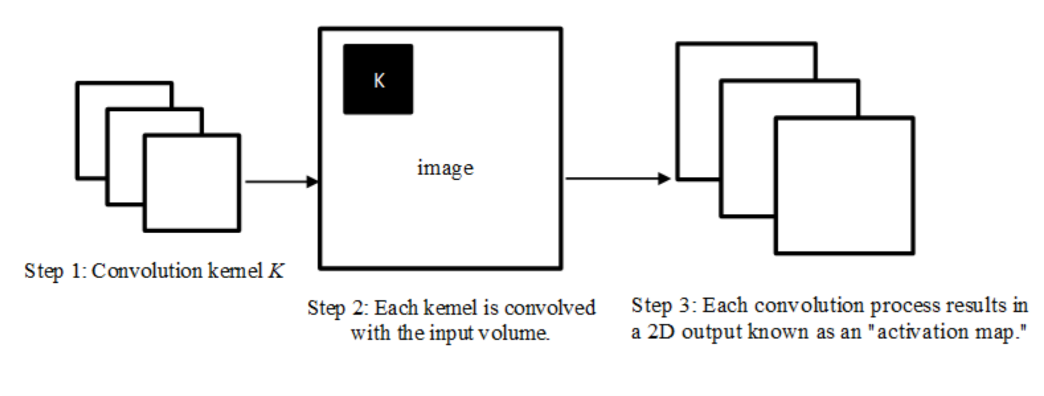

Figure 1: The convolution operation

2.2.2. Max Pooling Layer

The pooling layer compresses the input feature graph of 148×148×32, which on the one hand makes the feature graph smaller and simplifies the network computing complexity. On the one hand, feature compression is carried out to extract the main features. The size of the Pooling kernel is set to 2×2 and its move step is 2. We use Max Pooling to conduct the pooling operation, search for the maximum value in each 2×2 area, and the step size of each retrieval is 2, and finally extract the main feature 74×74×32 from the original feature map, as shown in Figure 2:

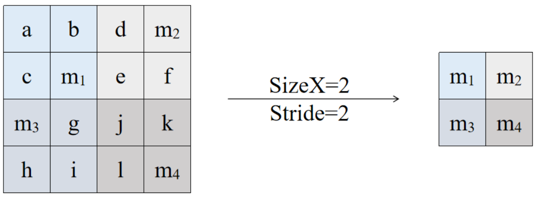

Figure 2: Max pooling operation.

2.2.3. Dropout Layer and Softmax Output Layer

After the previous training, the neural network is prone to overfitting problems, that is, after the training, the weight of the neural network matches the training sample too much, so the performance in processing the new sample is mediocre. To avoid such problems, we randomly discard a certain percentage of data during training. Here, we set the discard ratio for each Dropout layer to 0.1. The Dropout layer discards a random activation parameter set in this layer and sets these activation parameter sets to 0 with a probability of 0.1 to provide appropriate classification or output for a specific sample. This mechanism will ensure that the neural network does not overmatch the training sample, helping to alleviate the overfitting problem.

The Softmax layer converts the output result of the neural network and presents the output result in the form of probability. A picture to be classified is put into the model. Among the probabilities output by the Softmax classification layer, the label corresponding to the maximum probability is the label of this graph to be classified.

$$\sigma_i(z) = \frac{e^{z_i}}{\sum_{j=1}^{m} e^{z_j}} \quad (5)$$

$$z_i = w_i x + b \quad (6)$$

$$\sum_{i=1}^{j} \sigma_i(z) = 1 \quad (7)$$

The working principle of the Softmax layer is shown in Figure 3:

ISSN 2616-5775 Vol. 6, Issue 2: 16-22, DOI: 10.25236/AJCIS.2023.060203

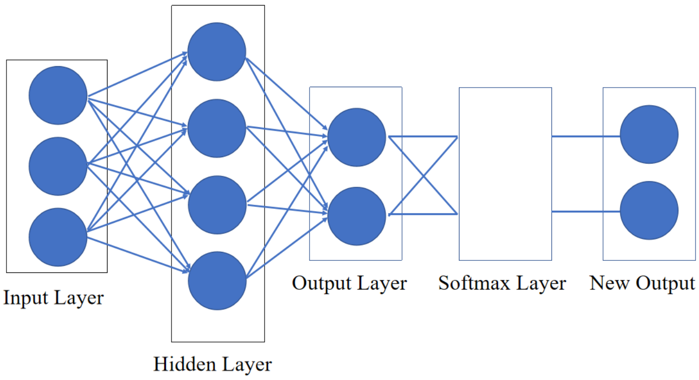

Figure 3: Working Principle Diagram of the Softmax Layer

2.3. A Complete Overview of Model Building

After constructing four rounds of the Convolution layer, Pooling layer, and Dropout layer alternately, the final output result is 7×7×64. The Flatten function is used to flatten the multidimensional data 7×7×64 into one-dimensional data 3136. In the fully connected layer, the number of units in the first layer is 128 and the activation function is Relu; the number of units in the second layer is 4, and the activation function is changed to Softmax, which is known as the Softmax output layer.

The overall structure is shown in Table 2:

Table 2: Model overview diagram

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 148, 148, 32)	896
max_pooling2d (Maxpooling 2D)	(None, 74, 74, 32)	0
dropout (Dropout)	(None, 74, 74, 32)	0
conv2d_1 (Conv2d)	(None, 72, 72, 32)	9248
max_pooling2d_1 (Maxpooling 2D)	(None, 36, 36, 32)	0
dropout_1 (Dropout)	(None, 36, 36, 32)	0
conv2d_2 (Conv2d)	(None, 34, 34, 64)	18496
max_pooling2d_2 (Maxpooling 2D)	(None, 17, 17, 64)	0
dropout_2 (Dropout)	(None, 17, 17, 64)	0
conv2d_3 (Conv2d)	(None, 15, 15, 64)	36928
max_pooling2d_3 (Maxpooling 2D)	(None, 7, 7, 64)	0
dropout_3 (Dropout)	(None, 7, 7, 64)	0
flatten (Flatten)	(None, 3136)	0
dense (Dense)	(None, 128)	401536
dropout_4 (Dropout)	(None, 128)	0
dense_1	(None, 4)	516

Total params: 467,620
Trainable params: 467,620
Non-trainable params: 0

The overall neural network diagram is shown in Figure 4:

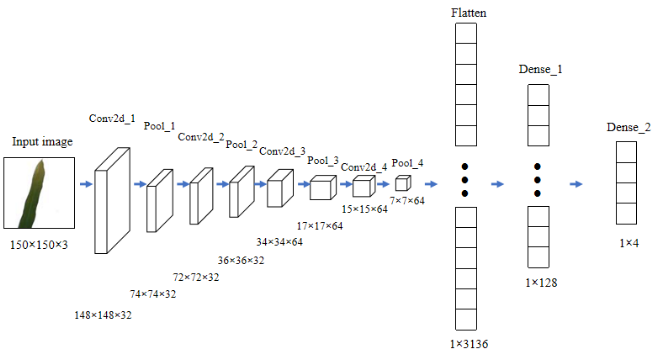

Figure 4: Schematic diagram of convolutional neural network

ISSN 2616-5775 Vol. 6, Issue 2: 16-22, DOI: 10.25236/AJCIS.2023.060203

3. Optimizer Comparison

In the process of model optimization, it is mainly to change the optimizer function to realize the optimization of the whole model. In this experiment, the Adam algorithm was used first, and then the Nadam algorithm was used to improve the Adam algorithm.

The Adam algorithm, namely Adaptive Moment Estimation, is a combination of the AdaGrad and Momentum methods in name, but its essence combines the RMSprop method and the Momentum method.

The RMSProp algorithm is an improvement of the AdaGrad algorithm. The AdaGrad algorithm considers all historical gradients, and it will happen when it should change rapidly. Because considering the past historical gradient problem, there is still no rapid decline. The RMSProp algorithm solves the limitations of the AdaGrad algorithm by using relatively recent historical gradient data, and will adaptively change its learning rate.

$$S_{(t)} = \beta S_{(t-1)} + (1-\beta)\Delta W_{(t)i} \cdot \Delta W_{(t)i} \quad (8)$$

$$W_{(t-1)i} - \frac{\eta}{\sqrt{S_{(t)}+\varepsilon}} \cdot \Delta W_{(t)i} W_{(t)i} \quad (9)$$

The Momentum method optimizes the oscillation of the Newton method in some cases. The Momentum method reduces the oscillation in the axial direction and speeds up the descent in the gradient direction by referring to historical data.

The combined Adam algorithm can comprehensively consider the mean and variance of the gradient and calculate the updated step size. To achieve fast and efficient calculation, it is suitable for a variety of situations and automatically realizes the adjustment of the learning rate.

$$V_{(t)} = \beta_1 V_{(t-1)} + (1-\beta)\Delta W_{(t)i} \quad (10)$$

$$S_{(t)} = \beta_2 S_{(t-1)} + (1-\beta_2)\Delta W_{(t)i} \cdot \Delta W_{(t)i} \quad (11)$$

$$W_{(t)i} = W_{(t-1)i} - \frac{\eta}{\sqrt{S_{(t)}+\varepsilon}} \cdot V_{(t)} \quad (12)$$

Nadam further combines the Nesterov and adam algorithms. Compared with the Momentum method, the Nesterov algorithm introduces the parameter of the derivative of the target in the calculation, so that the Nesterov algorithm can "predict" the direction of the next gradient descent in a certain program so that it can converge more quickly. It is reflected in the formula for calculating the gradient:

$$g_t = \nabla f(\theta_t - \frac{\alpha m_{t-1}}{\sqrt{V_{t-1}}}) \quad (13)$$

In this experiment, the Adam algorithm was used first, and then the Nadam algorithm was used.

4. Conclusion

4.1. Data Introduction

The dataset we used contained both training and testing data. The training folder contains 5390 images and the test folder contains 400 images. The dataset consists of four classes of nitrogen-deficient rice crop leaf images matched with leaf color maps taken from Sambalpur University. It contains four sub-folders, each corresponding to the nitrogen deficiency level of the leaf color chart.

We experimentally demonstrated the practical performance of a CNN-based model for image classification of rice nitrogen deficiency. By feeding the preprocessed dataset to the CNN, adding the network structure layer by layer using appropriate methods, and then optimizing the objective function, we obtained the loss function and the accuracy with increasing epochs.

4.2. Result Assessment

We performed geometric transformation operations such as flipping, translation, and rotation on existing pictures and adjusted color transformations such as brightness, chroma, and contrast. We used the augmented image training set for training. In deep learning, loss and accuracy are two important indicators; loss refers to the gap between the model prediction result and the real result; typically, we want the model to be as small as possible; accuracy refers to the accuracy of the model, that is, the degree

ISSN 2616-5775 Vol. 6, Issue 2: 16-22, DOI: 10.25236/AJCIS.2023.060203

to which the model prediction result matches the real result; usually, we expect a higher level of accuracy. When solving deep learning problems, we used the training data to calculate the loss and accuracy of the model and applied the optimization algorithm to continuously adjust the model parameters so that the loss and accuracy of the model on the training data are getting better and better.

Generally speaking, the function of our prediction can be more satisfactory. We used the train set to train our models and recorded the loss and accuracy of each epoch. We made comparisons between them, and finally we chose the best optimizer according to the results.

In this experiment, we used the augmented image training set for training. The Adam algorithm was used first, and then the Nadam algorithm was used. When we set the training epoch to 20 rounds, the training accuracy and loss curves obtained are compared in Figure 5:

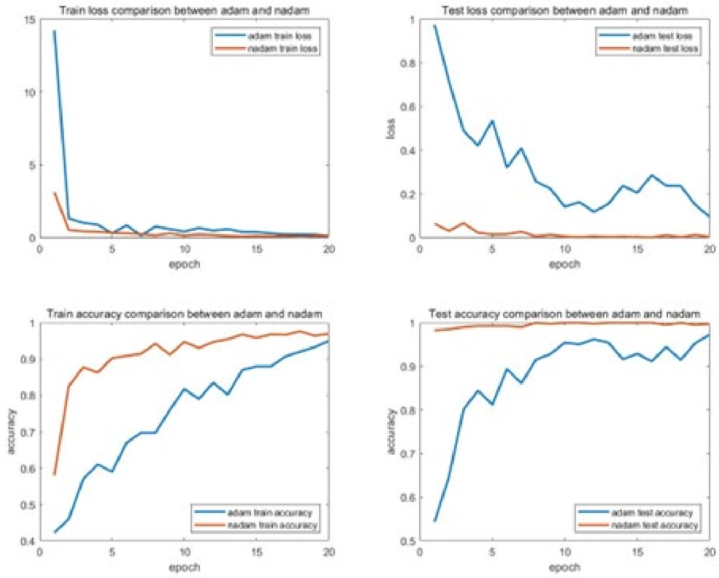

Figure 5: Before optimization training and testing set model accuracy and loss curve

By using the CNN we introduced, the model with the Nadam optimizer can achieve relatively high accuracy in practice. Using the CNN we introduced, the accuracy of the training set can reach 0.9500, and after 20 epochs, the accuracy of the test set is about 0.9727. This indicates that the model not only converges quickly but also has good assembly performance. Moreover, the performance was the best in terms of accuracy and convergence speed. As we can see in the figure above, the accuracy of the model continues to increase with some fluctuations, and the model follows the loss on the training set. The decrease in accuracy gradually increases, the overall trend loss decreases, and accuracy rises, which means that this model performs better in the recognition of the optimized picture.

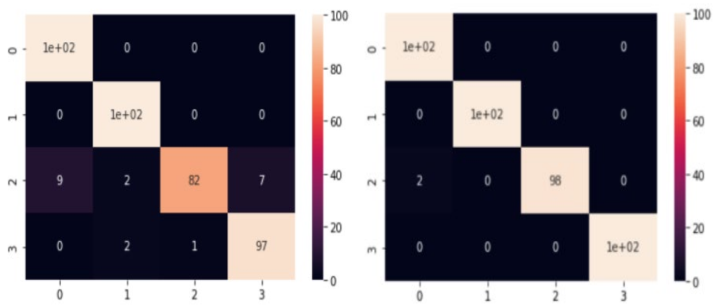

Figure 6: The original confusion matrix and post-optimization confusion matrix

The original confusion matrix and the optimized confusion matrix are shown in Figure 6. By calculating the accuracy of the original confusion matrix and the optimized confusion matrix, the value of accuracy ranges from 94.75% to 99.5%, and the analysis shows that the classification of most images is correct before and after optimization, and the optimized accuracy is more accurate. In future studies, we will further evaluate the influence of other parameters on the model and pre-train some popular

ISSN 2616-5775 Vol. 6, Issue 2: 16-22, DOI: 10.25236/AJCIS.2023.060203

models to achieve high accuracy.

References

[1] P. K. Sethy, N. K. Barpanda, A. K. Rath, and S. K. Behera, "Nitrogen Deficiency Prediction of Rice Crop Based on Convolutional Neural Network," J Ambient Intell Human Comput, vol. 11, no. 11, pp. 5703–5711, Nov. 2020.

[2] R. L. Mulvaney, S. A. Khan, and T. R. Ellsworth, "Synthetic Nitrogen Fertilizers Deplete Soil Nitrogen: A Global Dilemma for Sustainable Cereal Production," J. Environ. Qual., vol. 38, no. 6, pp. 2295–2314, Nov. 2009.

[3] L. S. Chen and K. Wang, "Diagnosing of rice nitrogen stress based on static scanning technology and image information extraction，" J Soil Sci Plant Nutr, vol. 14, no. 2, pp.382-393, Nov. 2014.

[4] W.-H. Yang, S. Peng, J. Huang, A. L. Sanico, R. J. Buresh, and C. Witt, "Using Leaf Color Charts to Estimate Leaf Nitrogen Status of Rice," AGRONOMY JOURNAL, vol. 95, 2003.

[5] Arnal Barbedo, Jayme Garcia. "Digital image processing techniques for detecting, quantifying and classifying plant diseases." SpringerPlus vol.2 no.1,pp:1-12,2013.

[6] Zhang Jian and Zhang Wei, "Support vector machine for recognition of cucumber leaf diseases," in 2010 2nd International Conference on Advanced Computer Control, Shenyang, China, pp. 264–266, 2010.

[7] A. Asfarian, Y. Herdiyeni, A. Rauf, and K. H. Mutaqin, "A computer vision for rice disease identification to support Integrated Pest Management," Crop Protection, vol. 61, pp. 103–104, Jul. 2014.

[8] Y. Chen, J. Wang, R. Xia, Q. Zhang, Z. Cao, and K. Yang, "RETRACTED ARTICLE: The visual object tracking algorithm research based on adaptive combination kernel," J Ambient Intell Human Comput, vol. 10, no. 12, pp. 4855–4867, Dec. 2019.

[9] F. Moslehi and A. Haeri, "An evolutionary computation-based approach for feature selection," J Ambient Intell Human Comput, vol. 11, no. 9, pp. 3757–3769, Sep. 2020.

[10] S. Misra and R. H. Laskar, "Development of a hierarchical dynamic keyboard character recognition system using trajectory features and scale-invariant holistic modeling of characters," J Ambient Intell Human Comput, vol. 10, no. 12, pp. 4901–4923, Dec. 2019.