Research Article

Research on Community Risk Prediction Model and Management Based on Deep Learning

Weida Yin

1School of Public Administration, China University of Mining and Technology, Xuzhou 221008, China
2Law & Politics School, Suqian University, Suqian 223800, China

Correspondence should be addressed to Weida Yin; 21040@squ.edu.cn

Received 31 December 2021; Revised 7 February 2022; Accepted 21 February 2022; Published 11 March 2022

Academic Editor: Zhiguo Qu

Copyright © 2022 Weida Yin. 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.

The influencing factors of community risk are complex. For the low accuracy of traditional prediction model, a multichannel convolutional neural network community risk prediction model is proposed by improving convolutional neural network of deep learning. First of all, in the community risk prediction model, the structure of multichannel input convolutional neural network is selected. Then, add it into the full connection layer. Subsequently, the DenseNet layer is added to establish connections between different network layers. Finally, the receptive field is improved, and the gradient disappearance is solved. Thus, the prediction accuracy of model is improved. Compared with the traditional model, the proposed multichannel convolutional neural network model has better prediction accuracy. In addition, it performs better on the three indicators, namely, correlation coefficient $R$, coefficient of determination $R^2$, and mean square root error RMSE. Compared with the commonly used LSTM model and logic regression model, the proposed model also has certain advantages, which is more suitable for community risk prediction.

1. Introduction

Community is the basic unit of people’s life. Community risk is not only related to the safety of residents in the community but also related to the social public stability. The important measures to ensure the stability of social security are to strengthen the ability of community risk prevention and response and raise the safety consciousness of community residents. However, with the development of urbanization, the risks faced by the community in the process of development are becoming more and more complex. Therefore, the harm caused by community security risk is minimized by monitoring community security risk.

Early social risk early warning mainly evaluates community security through Dempster-Shafer theory to realize community risk early warning. For example, Yin and Krzysztof et al. used Dempster-Shafer theory to study the safety risks of megacities community. A system of urban community safety evaluation indicators is constructed based on the static and dynamic factors. However, there are still problems for combining the static and dynamic factors to predict, such as insufficient content, insufficient index, and inaccurate model. Fortunately, the problems can be eliminated, which makes the evaluation of urban community safety more effective [1, 2]. Patel et al. described the association of social risk and exclusion factors with health outcomes for one of the largest populations in MENA [3]. At the same time, Patel et al. found that people with higher social risk factors reported poor health. It is beneficial to the screening and referral models of the social needs of the MENA subpopulation in the United States. As can be seen, deep learning has been widely used to predict risk, which provides new ideas for community risk prediction. For example, Aditya et al. explored the details of cell phenotypes based on the advantages of deep learning in the recognition, analysis and prediction of visual phenotypes, which provides a reference for the application of the details of cell phenotypes in important biological problems [4]. Ozge and Zhang et al. realized high-resolution fuse classification based on recursive graph convolutional neural network [5, 6]. Dong et al. used handheld Raman spectrometers in public to collect hazardous chemicals data from public safety. At the same
time, Dong et al. used the structure of convolutional neural network to determine hazardous chemicals in public safety [7]. Based on convolutional neural network, Frederic et al. realized the prediction of possible burning sites in wasteland through numerical simulation of the spread of wasteland fire. Therefore, a new application of short-term fire risk mapping is opened [8].

The above research shows that convolutional neural network has certain advantages in risk prediction in various fields, but there is still a problem of improving the prediction accuracy. Thus, based on convolutional neural network, and combined with community risk-related features, a multichannel convolutional neural network prediction model is proposed, where the single-channel input is improved to multichannel. In addition, DenseNet layer is added to establish connections to different network layers.

2. Basic Methods

2.1. Introduction to Convolutional Neural Networks. Convolutional neural network is a deep learning algorithm with deep structure. It is widely used in computer recognition and image recognition. Traditional convolutional neural network consists of five layers, namely, input layer, convolutional layer, pooling layer, fully connected layer, and output layer. The structure is as shown in Figure 1. Among them, the convolution layer is responsible for extracting features through convolution operation. And in mathematical calculus, the specific calculation formula is as follows [9]:

$$s(t) = (x * w)(t) = \int x(a)w(t-a)da. \quad (1)$$

In the formula, x represents the input, and w represents the convolution kernel. In addition, a represents the integral, t represents the function displacement, and t represents the feature mapping. The discrete behavior of convolution is

$$s(i,j) = (I * K)(i,j) = \sum_{m,n} I(m,n)K(m-j,n). \quad (2)$$

Since convolution is variable, the Formula (2) is expressed as [10]

$$s(i,j) = (I * K)(i,j) = \sum_{m,n} I(i-m,j-n)K(m,n). \quad (3)$$

The pooling layer is responsible for downsampling the output feature mapping of the convolutional layer to further extract features and reduce weight parameters. There are average pooling and maximum pooling.

The activation function is usually connected. There are three common activation functions, namely, Tanh function, RelU function, and Sigmoid function. Sigmoid function is shown in formula (4). Tanh function is shown in formula (5). And the simplified calculation is shown in formula (6). RelU function is shown in formula (7). Compared with Sigmoid function and Tanh function, RelU function has certain advantages in solving gradient disappearance [10, 11].

Therefore, RelU function is the activation function in this paper.

$$\sigma_s = \frac{1}{1 + e^{-t}}, \quad (4)$$

$$\tanh = \frac{e^t - e^{-t}}{e^t + e^{-t}}, \quad (5)$$

$$\tanh (x) = \frac{2}{1 + e^{-2x}} - 1, \quad (6)$$

$$\text{ReLU} = \max (0, x). \quad (7)$$

To prevent cell inactivation, usually when $x < 0$, a non-0 slope is introduced into the RelU function, namely, [12]

$$g(x) = \max (0, x) + a \min (0, x). \quad (8)$$

When $a = -1$, namely, $g(x) = |x|$ represents the absolute value correction method [13].

The recognition and classification of information are realized through the interaction between different layers. This network has the characteristics of parameter sharing, feature invariability, and sparse connection [14]. However, it can only accurately predict a single input feature. The community risk of the research object in this paper has many influencing factors. Furthermore, the input network has many related feature factors. Therefore, a multichannel convolutional neural network model is proposed to predict community risk.

2.2. Introduction to Multichannel Convolutional Neural Networks. In the convolutional neural network model, multichannel input is adopted, and DenseNet layer is added to establish connections between different network layers, as can be seen that the sensory field and prediction accuracy are improved. The structure is shown in Figure 2, where the left part is the traditional convolutional neural network, and the right part is the improved multichannel convolutional neural network.

There are eight parts for the multichannel convolutional neural network, namely, input layer, convolutional layer, pooling layer, flatten layer, fully connected layer, DenseNet, dropout layer, and output layer. Input layer consists of four feature inputs. Each feature input is convoluted by two convolution layers, pooled by a maximum pooling layer, and flattened by a flatten layer. Then, it passes fully connected layer and DenseNet layer. Each component layer in the DenseNet layer uses the BN and RelU functions and then uses a four-channel output feature to conduct $3 \times 3$ convolution and through the three-layer Dense block. Meanwhile, average pooling operation is performed, and a softmax classifier is added. Then, after passing through the dropout layer with a dropout value of 0.5, the final prediction result is output by the output layer.
3. Community Risk Prediction Model Based on Multichannel Convolutional Neural Network

The community risk prediction model of multichannel convolutional neural network is constructed as follows:

1. Data collection: collect and organize the factors affecting community risk and screen out the characteristic factors with strong correlation after preprocessing such as data cleaning and normalization.

2. Dataset division: the data is divided into training set and testing set in the ratio of 4:1.

3. Construction of multichannel convolutional neural network model: input layer: each input variable corresponds to an input layer $L_i$, and one-dimensional convolution kernel is used to extract input variable features for each layer $L_i$. The input layer is a vector $V_i = [D_1, D_2, D_3, D_4, D_5, D_6, D_7]$ for a variable week, which represents a three-dimensional matrix with the shape of $1 \times 7 \times 1$, and the single input shape is $(1, 7, 1)$. Convolution layer: there are 32 convolution kernels in one-dimensional convolution layer, and the size is $3$. RelU function is activation function. Pooling layer: the maximum pooling layer size is 2. Flatten layer: the flattened feature obtained by the flattened layer is $F_i$. Each $F_i$ is spliced to obtain a complete flattened eigenvalue $F = [F_1, F_2, F_3, F_4]$. Fully connected layer: neurons in full connection layer $D_2$ is 100, and the activation function is RelU function. Output layer: the number of neurons is 1, which corresponds to the data value on the eighth day after one week.

4. Model training: use training dataset to train the model. Mean square error (MSE) is used as loss function, and Adam is used as optimizer. The number of iterations is 20, and batch-size is 4 [15]. When the model output value and the MSE of observation value do not change or reach the maximum number of iterations, save the model.

5. Based on the above saved model, testing dataset is used to predict and output prediction results. Comparing the output prediction results with the actual value, the accuracy of the model is calculated; that is, the prediction of community risk is completed.
The above process can be illustrated in Figure 3.

4. Simulation Experiment

4.1. Data Sources and Preprocessing

4.1.1. Data Sources. Community risk is related to its basic information, case information, climate information, community population information, and many other factors. Therefore, in order to realize community risk prediction and management, community risk data sources should be obtained from many aspects. In this simulation reference [16], the security cases, climate data and second-hand housing price data of a community in a third-tier city in China from January 2017 to December 2019 are selected as experimental data. Among them, the data of public security cases are provided by the public security bureau. Through desensitization of the data and screening according to the address of the case, a total of 12,000 data of community security cases data are obtained. Climate data are counted on a daily basis, and a total of 1,095 pieces of data are obtained. In addition, the second-hand housing price and the number of community security cases are shown in Figure 4.

4.1.2. Data Preprocessing

(1) Missing Value Preprocessing. There may be missing values in the experimental data. Avoiding the influence of missing value on model prediction effect, the data, such as the small amount of data and the weather that is easy to be queried, is performed the artificial fill processing. For the community public security cases with no influence on the prediction results and the large amount of data, the early-warning data is deleted. For the missing data of the number of community security cases of important research objects, the moving window mean filling method is adopted for processing, and the mean value of the cases near the missing data is used as the missing data case loads, which can be expressed as the formula as [17]

\[ X_m = \frac{1}{6} (X_{m-3} + X_{m-2} + X_{m-1} + X_{m+1} + X_{m+2} + X_{m+3}), \]  

where \( X_{m-n} \) stands for the number of cases near the missing value; \( X_m \) represents the missing value filling the number of cases.

(2) Normalization. Considering the difference of experimental data magnitude, they are normalized, and the specific function is expressed as Formula (10) [18]:

\[ y = \frac{x - \text{Min}}{\text{Max} - \text{Min}}. \]  

Here, \( x \) stands for the original data value, \( y \) is the normalized data value, \( \text{Max} \) represents the maximum value of data, and \( \text{Min} \) represents the minimum data value.

In addition, considering that not all values can be mapped to the interval [0, 1], for practical application, all target mapping interval is assumed to \([\alpha, \beta]\), and the result of original data mapping to interval \( \alpha, \beta \) is [19]

\[ Y = \alpha + \frac{x - \text{Min}}{\text{Max} - \text{Min}} \times (\beta - \alpha). \]  

4.2. Feature Extraction

4.2.1. Feature Screening. Extracting features is the key to construct community risk prediction model. In order to screen out the feature factors most relevant to community risk prediction, Pearson correlation coefficient is improved for the influencing factors analysis. The original calculation formula of Pearson correlation coefficient is shown in formula (12) [20]:

\[ R = \frac{\sum_i^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_i^n (Y_i - \bar{Y})^2} \sqrt{\sum_i^n (X_i - \bar{X})^2}}. \]  

In the formula, there are two related variables, namely, \( X \) and \( Y \). \( X_i \) is the community security cases, and \( Y_i \) is the screened characteristic data. Moreover, the mean values of \( X_i \) and \( Y_i \) are respectively \( \bar{X} \) and \( \bar{Y} \) respectively. And the correlation between \( X \) and \( Y \) is \( R \).

The number of moving average community security cases is adopted to obtain the smooth number \( M \). Moreover, using \( M \) to replace \( X \), the improved Pearson backbone coefficient could be obtained. The correlation smoothing coefficient \( R_m \) can be expressed as [21]

\[ R_m = \frac{\sum_i^n (X_i - M_i)(Y_i - \bar{Y})}{\sqrt{\sum_i^n (Y_i - \bar{Y})^2} \sqrt{\sum_i^n (X_i - M_i)^2}}, \]  

where the value range of \( R_m \) is [-1, 1]. When the value is within [-1, 0], variables are negative correlated. When its value is within [0, 1], variables are positive correlated. The larger the value of \( R_m \), the higher the correlation degree is. When the correlation is less than 0.05, there is no significant correlation between variables [22].
4.2.2. Correlation Analysis of Feature Factors. The proposed model is to predict the daily community security case number. When analyzing feature correlation, the data that never changes in a short time is ignored, such as residential population density. The public security cases, noise reduction cases, second-hand housing price, climate, and vacation data are selected to analyze the correlation. Pearson correlation is used to test each influencing factor, and the results are shown in Table 1. There is a positive correlation between the second-hand housing price and the number of noise reduction security cases, with a correlation coefficient of 0.44678 and a great correlation. There is a negative correlation between climate, vacation, and noise reduction cases, with correlation coefficients of -0.1512 and -0.3027, respectively, showing a significant correlation. This shows that the second-hand housing price, climate, holidays, and the number of public security cases have an important impact on the community security risk. Therefore, this paper selects the above four types of feature data as input of the model.

4.3. Evaluation Indicators. Root mean square error (RMSE) is used to evaluate the prediction effect of the model, as shown in formula (14) [23]. The smaller RMSE value is, the better the predicted value is. The larger the value is, the closer the predicted value is to the actual value, and the better the robustness of the model is.

\[
\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2},
\]

(14)

Formula (14) shows that \(y_i\) is the true value. \(\hat{y}_i\) is the predicted value.

\[
r(X, Y) = \frac{\text{Cov}(X, Y)}{\sqrt{\text{Var}[X]\text{Var}[Y]}},
\]

(15)

\[
R^2 = 1 - \frac{\text{SS}_\text{res}}{\text{SS}_\text{tot}} = \frac{\text{SS}_\text{reg}}{\text{SS}_\text{tot}},
\]

(16)

\[
\hat{y} = \frac{1}{n} \sum_{i=1}^{n} y_i,
\]

(17)

\[
\text{SS}_\text{tot} = \sum_{i=1}^{n} (y_i - \hat{y})^2,
\]

(18)

\[
\text{SS}_\text{res} = \sum_{i=1}^{n} (y_i - \hat{y})^2, \quad \text{SS}_\text{tot} = \sum_{i=1}^{n} (y_i - \hat{y})^2, \quad \text{SS}_\text{reg} = \sum_{i=1}^{n} (f_i - \hat{y})^2, \quad \text{SS}_\text{tot} = \sum_{i=1}^{n} (y_i - \hat{y})^2.
\]

(19)
Formula (15) shows that $\text{Cov}(X, Y)$ represents the covariance of $X$ and $Y$, which can be calculated by formula (20) [25]. $\text{Var}[X]$ represents the variances of $X$, which can be calculated by formula (21).

\[
\text{Cov}(X, Y) = E\{[X - E(X)][Y - E(Y)]\},
\]

\[
\text{var}(X) = \frac{\sum_{i=1}^{n} (X_{i} - \bar{X})(X_{i} - \bar{X})}{n - 1}.
\]

4.4. Parameter Settings. The final prediction result of convolutional neural network will affect model parameters. To obtain best prediction results, control variable method is used to determine convolutional layers, convolutional kernels, activation function, optimizer, and pooling operation mode of pooling layer. The model is set for 20 iterations. And its loss function is the mean square error, epoch = 20, batch – size = 4, and dropout = 0.5.

4.4.1. Determination of the Number of Convolutional Layer. The proposed model is constructed with different convolutional layers. And the community risk security cases are predicted. The results are shown in Table 2. When convolutional layers are 2, correlation coefficient and coefficient of determination are the largest, which are 0.9694 and 0.9389, respectively. And the root mean square error is the smallest, which is 0.8188. As can be seen that, when convolutional layers are 2, the prediction effect of model has a small error with the real value, and the prediction effect is better. Therefore, the number of convolutional layer is set to 2.

4.4.2. Determination of the Number of Convolution Kernels. Convolution kernels are responsible for feature extraction. The more the convolution kernels is, the more fully the model extracts features. However, too many convolution kernels will lead to overfitting, and the prediction accuracy of the model will be reduced. Therefore, it is necessary to determine the number of model convolution kernels. In addition, different numbers of convolution kernels are set up to build models and make predictions. The results are as shown in Table 3. When convolution kernels are 32, the correlation coefficient and coefficient of determination are the largest, which are 0.9697 and 0.9389, respectively. Meanwhile, the root mean square error is the smallest, which is 0.8188. Thus, convolution kernels are 32 in this paper.

4.4.3. Determination of Activation Function. The common activation functions are Sigmoid, Tanh, and ReLU functions. Different activation functions are selected to build models and make predictions. The results are shown in Table 4. When ReLU function is the activation function, the model performs better.

4.4.4. Optimizer Determination. Common model parameter optimizers include Adam and RMSprop. Different optimizers are selected to optimize the model parameters, and the model prediction results are obtained as shown in Table 5. When Adam is used as the parameter optimizer, the model performs better. Therefore, Adam is selected as the model parameter optimizer in this paper.

### Table 2: Comparison of model prediction results of different number of convolutional layer.

| The number of convolutional layer | $R$ | $R^2$ | RMSE |
|-----------------------------------|-----|-------|------|
| 1                                 | 0.9635 | 0.9300 | 0.8800 |
| 2                                 | 0.9694 | 0.9389 | 0.8188 |
| 3                                 | 0.9685 | 0.9342 | 0.8455 |

### Table 3: Comparison of model prediction results of different number of convolution kernel.

| The number of convolution kernel | $R$ | $R^2$ | RMSE |
|---------------------------------|-----|-------|------|
| 16                              | 0.9539 | 0.9112 | 0.8390 |
| 32                              | 0.9697 | 0.9389 | 0.8188 |
| 64                              | 0.9623 | 0.9210 | 0.8466 |

### Table 4: Comparison of model prediction results of different activation functions.

| Activation function | $R$ | $R^2$ | RMSE |
|---------------------|-----|-------|------|
| Sigmoid             | 0.8456 | 0.8265 | 1.1002 |
| Tanh                | 0.9470 | 0.8971 | 1.0688 |
| ReLU                | 0.9697 | 0.9389 | 0.8188 |

### Table 5: Comparison of model prediction results of different optimizer.

| Optimizer | $R$ | $R^2$ | RMSE |
|-----------|-----|-------|------|
| RMSprop   | 0.9589 | 0.9211 | 0.8790 |
| Adam      | 0.9697 | 0.9389 | 0.8188 |

### Table 6: Comparison of model prediction results of different optimizers.

| Pooling operation | $R$ | $R^2$ | RMSE |
|-------------------|-----|-------|------|
| Average pooling   | 0.9632 | 0.9377 | 0.8426 |
| Maximum pooling   | 0.9697 | 0.9389 | 0.8188 |

### Table 7: Model parameter settings.

| Parameter name      | Value |
|---------------------|-------|
| Convolutional layer | 2     |
| Convolutional filter| 32    |
| Activation function | ReLU  |
| Optimizer           | Adam  |
| Pooling             | Max pooling |
| Epoch               | 20    |
| Loss function       | MSE   |
| Batch_size          | 4     |
| Dropout             | 0.5   |

4.4.5. Determination of Pooling Operation. Pooling layer is an important network layer to prevent model overfitting. And the selection of pooling operation is particularly
important. Different pooling operations are used to construct the pooling layer and make predictions. The prediction results of the model are shown in Table 6. Compared with average pooling operation, when maximum pooling operation is selected, the model performs better.

Through the above experiments, the parameters of the constructed multichannel convolutional neural network model are finally determined as follows: there are 2 convolutional layers, there are 32 convolution kernels, the activation function is ReLU function, the optimizer is Adam, and the pooling operation is maximum pooling. The details are shown in Table 7.

4.5. Experimental Results

4.5.1. Model Verification. The improved effect of the proposed model on the convolutional neural network needs to be verified. Based on the above optimal parameter settings, the prediction results of model before and after the improvement are compared. The results are shown in Table 8. Compared with original model, the proposed model improves the prediction accuracy of model, and the structure of convolutional neural network model is improved. Moreover, the performance of proposed model is superior to the original model. Therefore, the improvement of convolutional neural network model is effective.

To verify the effectiveness of the model, a multichannel convolutional neural network prediction model is constructed under the optimal parameter setting. Furthermore, the prediction is made on the testing set. The comparison between the prediction result and actual observed value is shown in Figure 5. The predicted value curve of the model fits well with the actual observed value curve. There are many positions almost coinciding with each other, and the overall change trend is the same, which indicates that prediction effect of the proposed model is good.

For the prediction effect analysis of the model, correlation coefficient, coefficient of determination, and root mean square error are calculated on testing set. The results are shown in Table 9. The $R$ value is 87.77%, $R^2$ is 77.13%, and root mean square error is 0.8571, which meets prediction accuracy requirement. It indicates that the proposed model has good predictive ability. What is more, the correlation between predicted value and actual value is good.

4.5.2. Model Comparison. Experiment compares the prediction results of different models on the testing set, as shown in Table 10. Compared with the commonly used prediction model, such as LSTM model and logistic regression model,
the proposed multichannel convolutional neural network model has the largest determination error and the smallest root mean square error. Thus, the proposed model has certain advantages in community risk prediction, and its prediction performance is superior to LSTM model and logistics regression model.

5. Conclusion

To sum up, the proposed community risk prediction model improves the input of convolutional neural network. DenseNet layer is added to make the multichannel convolutional neural network model built. And the receptive field is improved. In addition, the connection between different network layers solves gradient disappearance problem, and the accuracy of model prediction is improved. Compared with the traditional model, namely, LSTM model and logistic regression model, the proposed multichannel convolutional neural network prediction model performs better on the RMSE index. It has certain advantages and is more suitable for community risk prediction. Meanwhile, it provides decision support for community security risk prevention. Although some research results have been obtained, there are still many shortcomings to be improved in the community risk prediction. The further research should be carried out from the following aspects: In the construction of multichannel convolutional neural network model, this paper mainly considers the model related to time series. However, the spatial factors in practical application also have a greater impact on community risk. Therefore, in the following study, the community spatial data should be incorporated into the analysis of community risk factors, so as to improve the practicality and generalization of the model.

Data Availability

The experimental data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares that he/she has no conflicts of interest regarding this work.

References

[1] K. Yin, H. Zhou, J. Chen, and C. Zhao, “The assessment of modern community security risks in megacities based on the Dempster–Shafer theory,” IOP Conference Series: Earth and Environmental Science, vol. 687, no. 1, pp. 012167–012176, 2021.
[2] J. Krzysztof and P. Jacek, “Reliability analysis of alarm signals transmitting systems used to monitoring buildings,” Journal of KONBIN, vol. 50, no. 3, pp. 1–20, 2020.
[3] M. R. Patel, M. Green, M. Tarig et al., “A snapshot of social risk factors and associations with health outcomes in a community sample of Middle Eastern and North African (MENA) People in the U.S.,” Journal of Immigrant and Minority Health, vol. 24, pp. 1–9, 2021.
[4] P. Aditya, D. Michael, and C. Caicedo Juan, “Image-based cell phenotyping with deep learning,” Current Opinion in Chemical Biology, vol. 65, pp. 9–17, 2021.
[5] O. F. Ozge and C. Mete, “Convolutional neural network analysis of recurrence plots for high resolution melting classification,” Computer Methods and Programs in Biomedicine, vol. 207, pp. 106139–106139, 2021.
[6] H. Zhang, H. Xu, X. Tian, J. Jiang, and J. Ma, “Image fusion meets deep learning: a survey and perspective,” Information Fusion, vol. 76, pp. 323–336, 2021.
[7] R. Dong, J. Wang, S. Weng, H. Yuan, and L. Yang, “Field determination of hazardous chemicals in public security by using a handheld Raman spectrometer and a deep architecture-search network,” Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy, vol. 258, pp. 119871–119871, 2021.
[8] A. Frédéric, M. Vivien, and F. Jean-Baptiste, “Emulation of wildland fire spread simulation using deep learning,” Neural Networks, vol. 141, pp. 184–198, 2021.
[9] Y. Liang, W. Peng, Z.-J. Zheng, O. Silvén, and G. Zhao, “A hybrid quantum-classical neural network with deep residual learning,” Neural Networks, vol. 143, pp. 133–147, 2021.
[10] H. Yu, Y. Zhou, R. Wang et al., “Associations between trees and grass presence with childhood asthma prevalence using deep learning image segmentation and a novel green view index,” Environmental Pollution, vol. 286, pp. 117582–117582, 2021.
[11] P. Priyadarshini and R. Kaushik, “Implicit adversarial data augmentation and robustness with Noise-based Learning,” Neural Networks, vol. 141, pp. 120–132, 2021.
[12] J. S. Han and K. C. Kwak, “Image classification using convolutional neural network and extreme learning machine classifier based on ReLU function,” The Journal of Korean Institute of Information Technology, vol. 15, no. 2, pp. 15–23, 2017.
[13] S. Byerly, L. R. Maurer, A. Mantero, L. Naar, G. An, and H. M. A. Kaaifarani, “Machine learning and artificial intelligence for surgical decision making,” Surgical Infections, vol. 22, no. 6, pp. 626–634, 2021.
[14] X. Zhao, X. Wang, and J. Sun, “Start-up demonstration tests with sparse connection,” European Journal of Operational Research, vol. 243, no. 3, pp. 865–873, 2015.
[15] P. Xing, J. Dong, P. Yu et al., “Quantitative analysis of lithium in brine by laser-induced breakdown spectroscopy based on convolutional neural network,” Analytica Chimica Acta, vol. 1178, pp. 338799–338799, 2021.
[16] Q. Zhou, W. Huang, X. Tian, Y. Yang, and D. Liang, “Identification of the variety of maize seeds based on hyperspectral images coupled with convolutional neural networks and sub-regional voting,” Journal of the Science of Food and Agriculture, vol. 101, no. 11, pp. 4532–4542, 2021.
[17] J. Xie, S. Chen, Y. Zhang, D. Gao, and T. Liu, “Combining generative adversarial networks and multi-output CNN for motor imagery classification,” Journal of Neural Engineering, vol. 18, no. 4, pp. 046026–046036, 2021.
[18] Z. Xinze, K. He, and Y. Bao, “Error-feedback stochastic modeling strategy for time series forecasting with convolutional neural networks,” Neurocomputing, vol. 459, pp. 234–248, 2021.
[19] X. Wang, X. Wang, W. Liu, Z. Chang, T. Kärkkäinen, and F. Cong, “One dimensional convolutional neural networks for seizure onset detection using long-term scalp and intracranial EEG,” Neurocomputing, vol. 459, pp. 212–222, 2021.
[20] J. Zhou, J. Gan, W. Gao, and A. Liang, “Image retrieval based on aggregated deep features weighted by regional significance.
and channel sensitivity," *Information Sciences*, vol. 577, pp. 69–80, 2021.

[21] L. Wu, X. Lin, Z. Chen, P. Lin, and S. Cheng, "Surface crack detection based on image stitching and transfer learning with pretrained convolutional neural network," *Structural Control and Health Monitoring*, vol. 28, no. 8, pp. e2766–e2766, 2021.

[22] H. Sharma, L. Drukker, A. T. Papageorghiou, and J. A. Noble, "Machine learning-based analysis of operator pupillary response to assess cognitive workload in clinical ultrasound imaging," *Computers in Biology and Medicine*, vol. 135, pp. 104589–104589, 2021.

[23] E. Rotunno, A. H. Tavabi, P. Rosi et al., "Alignment of electron optical beam shaping elements using a convolutional neural network," *Ultramicroscopy*, vol. 228, pp. 113338–113338, 2021.

[24] H. Zhu, H. Zeng, J. Liu, and X. Zhang, "Logish: a new nonlinear nonmonotonic activation function for convolutional neural network," *Neurocomputing*, vol. 458, pp. 490–499, 2021.

[25] N. Yuhao, A. S. Zamzam, and B. Adam, "Resampling and data augmentation for short-term PV output prediction based on an imbalanced sky images dataset using convolutional neural networks," *Solar Energy*, vol. 224, pp. 341–354, 2021.