EAI Endorsed Transactions
on e-Learning

Research Article EAI.EU

Gray Level Co-Occurrence Matrix and RVFL for Covid-19 Diagnosis

Wenhao Tang[1,*]

[1]School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo, Henan 454000, PR China

Abstract

As the widespread transmission of COVID-19 has continued to influence human health since late 2019, more intersections between artificial intelligence and the medical field have arisen. For CT images, manual differentiation between COVID-19-infected and healthy control images is not as effective and fast as AI. This study performed experiments on a dataset containing 640 samples, 320 of which were COVID-19-infected, and the rest were healthy controls. This experiment combines the gray-level co-occurrence matrix (GLCM) and random vector function link (RVFL). The role of GLCM and RVFL is to extract image features and classify images, respectively. The experimental results of my proposed GLCM-RVFL model are validated using K-fold cross-validation, and the indicators are 78.81±1.75%, 77.08±0.68%, 77.46±0.73%, 54.22±1.35%, and 77.48±0.74% for sensitivity, accuracy, F1-score, MCC, and FMI, respectively, which also confirms that the proposed model performs well on the COVID-19 detection task. After comparing with six state-of-the-art COVID-19 detection, I ensured that my model achieved higher performance.

Keywords: COVID-19, computed tomography, gray level co-occurrence matrix, random vector function link, K-fold cross-validation.

Received on 01 March 2023, accepted on 16 March 2023, published on 01 June 2023

doi: 10.4108/eetel.v8i2.3091

1. Introduction

According to reports, COVID-19 is an infectious disease that spread widely in late 2019. Patients infected with COVID-19 tend to have clinical manifestations of dry cough, malaise, and fever, and a few with upper respiratory and gastrointestinal symptoms such as nasal congestion, runny nose, and diarrhea. Pneumonia manifestations were present in severe and critically ill patients; however, no pneumonia manifestations were present in mildly ill patients. Some COVID-19 patients had mild symptoms, some without fever symptoms, and such patients mostly recovered after a week. Most patients are in good health after the cure, while a few are critically ill or even die. Severely and critically ill patients will have moderate to low or no significant fever during their illness.

Today, the most widely used COVID-19 test is RT-PCR, a nucleic acid detection method. However, RT-PCR nucleic acid detection methods have many problems in many countries, such as a fundamental shortage of detection reagents, insufficient testing capacity, and other complex issues. In addition, some experimental results confirmed that RT-PCR test results have a high rate of false negatives [1]. Chest CT is a helpful tool for diagnosing suspected cases of neo-coronavirus infection. However, in the specific circumstances of patients with other types of lung disease, CT-alone scans [2, 3] cannot identify the COVID-19 virus. Moreover, suppose the public COVID-19 dataset is not used correctly or has insufficient processing power. In that case, it can lead to modeling prediction bias, affecting model prediction performance [4].

In recent years, deep learning mechanisms have become an essential part of the computer vision field. As a branch of artificial intelligence, deep learning aims to allow machine learning models to improve their prediction

*Corresponding author. Email: wenhaotang@home.hpu.edu.cn

performance. In the COVID-19 suspected case diagnosis work [5, 6], multiple algorithms based on artificial intelligence can diagnose suspected cases more quickly and accurately compared to manual diagnosis by radiologists. When COVID-19 continues to spread, scientists and scholars in other fields are paying attention to efficient diagnostic techniques. I use the grayscale co-occurrence matrix to solve the problem of imperfect feature extraction from the dataset images. The GLCM is a simple and effective method to extract image texture features. It is equally a real-time image processing method. Therefore, the GLCM can be involved in COVID-19 suspected case diagnosis work fast enough [7].

RVFL has advantages in the field of data analysis. Nowadays, people in the research field with the following problems: during the neural network training process, the gradient-based algorithm network is deficient in speed; if the gradient algorithm is not optimized, it will fall into local minima. And for the backpropagation algorithm, there is a tendency to forget the old samples during training. Therefore, a hybrid network was proposed model that extracts image features with GLCM and classifies images with RVFL. The hybrid network model combines the advantages of both methods, making the model have better performance on multiple performance evaluation indicators. First, it has fewer parameters since RVFL has a single hidden layer. Therefore, RVFL is less prone to overfitting. Secondly, RVFL improves the generalization ability through invisible layer learning, effectively solving the problem of poor fitting to data outside the training set.

Section **Chyba! Nenašiel sa žiaden zdroj odkazov.** discusses the models in the world for diagnosing COVID-19, analyzes these models, and comments on their strengths and weaknesses. Section **Chyba! Nenašiel sa žiaden zdroj odkazov.** describes my preprocessing process for the dataset. Section **Chyba! Nenašiel sa žiaden zdroj odkazov.** describes the methods used to conduct my experiments. This section mainly describes the methods used for preprocessing, image classification, and dataset validation. It also shows some performance evaluation metrics. Section **Chyba! Nenašiel sa žiaden zdroj odkazov.** offers the analysis and discussion of my experimental results and compares the performance advantages and disadvantages of data analysis methods with state-of-the-art approaches. Section **Chyba! Nenašiel sa žiaden zdroj odkazov.** points out the conclusions of this paper and my outlook on future improvements of the model.

2. Related Work

Generally, the physician or specialist is responsible for diagnosing the CT scan images. This diagnosis is susceptible to various conditions not limited to the physical state and expertise of the physician or specialist. Therefore, efficient image recognition techniques generated by the combination of computer vision and artificial intelligence do not rely on manual diagnosis and can help diagnose suspicious cases of COVID-19.

Combining the network of RVFL can help diagnose other diseases. Sharma, et al. [8] proposed a deep non-iterative RVFL neural network for the early diagnosis of Alzheimer's disease. They used a pre-trained DNN to extract features from MRI images and used RVFL as a classifier. To solve the problem that MRI images may contain some outliers, the s-membership Fuzzy Activation Function (s-FAF) is used as the activation function to calculate the output of the hidden layer. The fuzzy function can map outliers to the range of precise values. Using RVFL as a classifier, they obtained the experimental results faster, mainly because RVFL performs non-iterative processing and has only one hidden layer. They also suggested that an excellent way of machine learning (ML) is to use support vector machines (SVM) for image classification.

RVFL is being promoted for use in the medical field to diagnose COVID-19. Hazarika and Gupta [9] combined the RVFL network with a one-dimensional discrete wavelet transform and then proposed a wavelet-coupled RVFL (WCRVFL) network to predict the spread of COVID-19. RVFL has a more significant advantage in image classification speed, while wavelet coupling can localize the features of the signal. To adapt to a more realistic prediction environment, they also used relu and sigmoid as the activation function of WCRVFL, which makes the prediction more accurate. It is experimentally confirmed that WCRVFL performs better than SVR and RVFL RELU models. For predicting the diffusion of COVID-19, wavelet coupling can predict the evolution trend of COVID-19 more accurately due to its characteristics.

Some issues with the COVID-19 datasets have led to poor performance of some networks trained and tested on these datasets. To address the lack of high-quality COVID-19 datasets, the Spatial Self-Attentive Network (SSA-Net), an encoder-decoder-based deep neural network, is used to segment lesions [10]. SSA-Net is competitive in performance compared to the state-of-the-art models in the COVID-19 diagnostic challenge. The semi-supervised iterative segmentation model of SSA-Net improves the learning ability in small and unbalanced training sets and can achieve higher performance. Moreover, the network can perform boosting and deconvolution without considering sequentially connected architectures, improving the decoding performance. SSA-Net uses few-shot learning to overcome the general lack of high-quality samples in the COVID-19 dataset. To cope with the problem of datasets containing incomplete data, Fang and Liang [11] also combined symptom-based ML models with a new learning mechanism called the Intensive Symptom Weight Learning Mechanism (ISW-LM) for early diagnosis of COVID-19. Goodfellow, et al. [12] combined two networks in adversarial states to create generative adversarial networks (GAN) for image and speech generation. Innovative networks based on GAN, such as the Progressive Growth Generative Adversarial Network (PGGAN) [13], and dual-model architectures using 3D

GAN and contrast learning methods [14], have been experimented on the COVID-19 dataset for classification.

2.1. ELM-BA

The extreme learning machine optimized by bat algorithm (ELM-BA) [15] utilizes the entropy of a 2D discrete wavelet transform (DWT) to extract features and uses ELM-BA as a classifier. ELM-BA was performed on a dataset containing 132 samples (114 are of pathological brains, and the others are of healthy controls), and its results were theoretically analyzed [15]. DWT has the function of multi-resolution analysis. The realization of this function depends on the wavelet transform of each input image by horizontal and vertical dimensions. The role of entropy is to reduce the number of features to cope with the problem of too many features that ELM-BA cannot solve.

Suppose $E(X)$ represents entropy, and X is a discrete and random variable. The values of X are in an infinite set. Then use $Y(X)$ to denote the probability function of X, and use $I(X)$ to reflect the uncertainty measure of a single event in the information. And then use K to denote the expectation operation. Then there is the functional equation:

$$E(X) = K[I(X)] = K[-\ln Y(X)] \qquad (1.)$$

If X is within the infinite dataset, then

$$E(X) = -\sum_i Y(X_i)\, log_b\, Y(x_i) \qquad (2.)$$

ELM has the advantages of fast learning and good generalization performance. The ELM algorithm randomly generates the connection weights between the input and hidden layers and randomly generates thresholds for the neurons in the hidden layer. And there is no requirement to adjust the above values during the model training. Only the number of hidden layer neurons is examined to obtain the unique optimal solution. However, since the model has not been developed and tested for other diseases, BA-ELM can only be used to identify pathological brains.

2.2. WEBBO

Like BA, Biogeography-Based Optimization (BBO) can also optimize the global optimal solution. Moreover, BBO also has a variational mechanism that BA does not have, which can prevent individuals in the algorithm from being constrained by a local extremum that is difficult to escape. Therefore, the model using BBO has a better performance in diagnosing suspected cases of COVID-19.

Yao [16] used Wavelet Entropy and Biogeography-Based Optimization (WE+BBO) for COVID-19 detection. BBO is a global optimization algorithm that avoids getting trapped in a local optimum. Many population-based optimization algorithms take a long time to run on CPUs. However, BBO converges faster because the goal of the BBO algorithm is to find the global-optimal solution, so it does not require unreasonable computation. BBO has features such as migration and mutation, where migration indicates probabilistic information sharing. Because each solution of BBO survives, the worse of them can receive many new features from the excellent solution. The wavelet transform can analyze unstable signals with multi-scale refinement and has adaptive capabilities. A wavelet transform is also an effective tool for analyzing and processing images and preserving image information, and it inherits the idea of localization of the Fourier transform. If the fundamental wavelet function $\varphi(t)$ with representation α is used as a different scale after the displacement τ, and the signal to be analyzed $x(t)$ is used as an inner product with $\varphi[(x-\tau)/\alpha]$, then

$$WT_x(\alpha,\tau) = \frac{1}{\sqrt{\alpha}}\int_{-\infty}^{+\infty} x(t)\varphi\left(\frac{t-\tau}{\alpha}\right)dt \qquad (3.)$$

In the above equation, α is called the scale factor, $\alpha > 0$. Both α and τ are continuous variables and the former acts as a stretch for the function $\varphi(t)$.

2.3. GLCM-SVM and GLCM-ELM

SVM and ELM are commonly used as image classifiers in machine learning. As a neural network that emerged later than SVM, ELM outperforms SVM in multi-classification. And ELM may have advantages over SVM in learning rate and generalization ability.

Chen [17] suggested the GLCM and SVM (GLCM-SVM), a hybrid network model for COVID-19 image classification. They utilize GLCM to transfer the image to a matrix, retain the matrix features to extract image features, and input it to SVM for image classification. Compared with the classifiers KNN and Naive Bayes, the network model they proposed has a higher performance. But GLCM-SVM needs to be strengthened in terms of accuracy.

The hybrid network model combining GLCM and ELM proposed by Pi [18] for diagnosing COVID-19 outperforms the network combining GCLM and SVM in several performance metrics. Because first, the ELM algorithm contains direct feature mapping, while the feature mapping of the SVM-based kernel method is indirect. Second, the input, implicit and output layers of ELM are connected, while SVM does not consider the connection of features in the layers of the neural network. Third, SVM classifies the data by constructing a hyperplane when solving, while the output layer of ELM has no error nodes and no such process. Fourth, ELM can solve multiclass classification directly, while SVM needs to convert multiclass categories into binary classification. Fifth, although the learning errors of ELM and SVM are comparable, the computational complexity of ELM is lower and faster than that of SVM, and the performance of GLCM-ELM has been experimentally confirmed to be more stable. However, there is room for further improvement in the accuracy and precision of this model.

2.4. WE-CSO and WE-GA

Cat swarm optimization (CSO) and genetic algorithm are common algorithms for optimizing the global optimal solution. Similar to the BA, the CSO converges slowly in the late iteration. The GA is scalable, easily combined with other algorithms, and has a simple search process.
Wang [19] also proposed wavelet entropy and cat swarm optimization (WE-CSO) to classify COVID-19 images, achieving the best performance among the comparable algorithms. CSO is also a global optimization algorithm containing both seeking and tracing modes. In the CSO algorithm, first, initialize the cat colony. Then divide the cats into two parts, and execute two patterns separately. The two modes are interacted by Mixture Ratio (MR) and update the colony after the interaction. If the optimization result of the cat colony achieves the goal, then the algorithm ends. Otherwise, the current optimal solution is calculated by the fitness function and retained; then, assign the cat again to perform seeking mode or tracing mode according to MR. Then according to this method, a predefined number of iterations are computed. The role of WE and CSO is to extract image features and optimize algorithms, respectively.

Wang [20] combined WE and genetic algorithm (WE-GA), and the proposed model WE-GA integrates the advantages of solid convergence and high robustness of the genetic algorithm. In addition, in the crossover process of the genetic algorithm, the genes with better performance can be retained, promoting the results close to the global optimal solution. They classify images using a feedforward neural network (FNN). FNN is an artificial neural network, as shown in Figure 1. It does not have cycles between connections and is all one-way propagation. And its network architecture contains hidden layers.

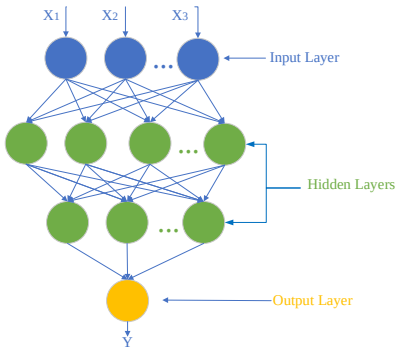

Figure 1. Structure of FNN.

They use 10-fold cross-validation to divide the training set and test set from the data set many times, which can reduce the randomness and improve the generalization ability. The cross-validation method also reduces the overfitting phenomenon. It can also address the issue of how to obtain as much information as possible from limited data. However, WE-GA has not learned on larger datasets. Therefore, there is still room to improve the model's performance due to insufficient image categories in smaller datasets.

2.5. SSA-Net

The spatial self-attention network consists of a feature encoder with self-attention learning, a feature re-extractor with spatial convolution, and a feature decoder. The use of SSA-Net is lesion segmentation of CT images. And it utilizes the semi-supervised learning approach. Instead of processing image features traditionally, Xiaoyan Wang connected the encoder layer to the four decoder layers. This approach allows them to perform upscaling and deconvolution regardless of the sequentially connected architecture, improving decoding performance [21]. As shown in Figure 2, after each of the four residual blocks of the encoder, a self-attention learning module is added, enhancing the model learning capability. And there is further a function of attention loss in the self-attentive learning module. The loss function plays a crucial role in the performance of the model.

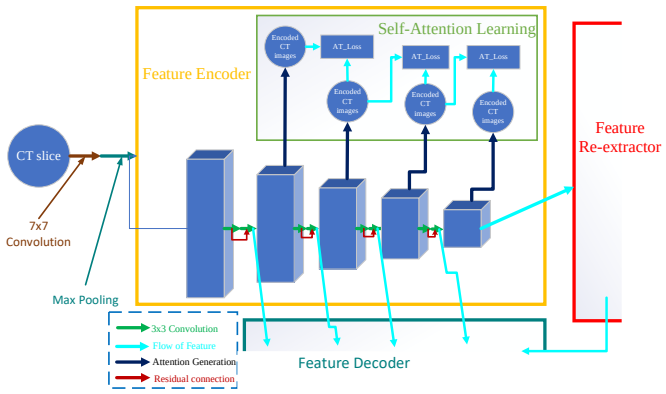

Figure 2. Structure of Feature Encoder.

Setting the correct loss function facilitates the model to exclude invalid data and focus on the right set of features in the data, accelerating the model convergence and improving the model performance. The loss function has an important impact on the training direction of calibration model prediction [22]. Datasets generally have a vast amount of data. So if the model learns only valid data, it can enhance the model training quality and improve the model training speed.

The feature map output from the fourth residual block is output to the feature re-extractor, where sequentially arranged modules perform the spatial convolution operation in the feature re-extractor. If $T_{i,j,k}$ denotes the element of a 3D tensor, and i, j, k indicate the channel, height, and width of indexes, respectively. Then L

represents the nonlinear activation function of ReLU, and T' denotes the update of the element, and $K_{m,i,n}$ denotes the weight between the channel m of the previous slice and the channel i of the current piece. Then, the spatial convolution function is

$$T'_{i,j,k} = \begin{cases} T_{i,j,k} & j = 1 \\ T_{i,j,k} + L\left(\sum_m \sum_n T'_{m,j-1,k+n-1} \times K_{m,i,n}\right) & j = 2, \dots, H \end{cases} \quad (4.)$$

Data extracted by the Feature Re-extractor are input to the feature decoder. The data information transferred in the feature encoder can skip the connection and input directly to the feature decoder, which facilitates the decoder to obtain more data from the encoder. Each decoder layer has a convolutional layer, and the kernel size of each layer is not entirely the same. To make the input image of SSA-Net the same size as the output image of SSA-Net, the Sigmoid function is finally used as the activation function to generate the segmentation result. The SSA-Net based on semi-supervised few-shot learning has excellent model performance and can be competent for the case of a small sample size. However, SSA-Net requires plausibly labeled samples as a dataset. Owing to the urgency of COVID-19 dissemination, large datasets, and annotations are sometimes missing. As a result, the performance of SSA-Net may be affected.

2.6. ISW-LM

The intensive Symptom Weight Learning Mechanism is a new learning mechanism with symptom-based machine learning. The machine learning (ML) method can analyze the importance of different disease symptoms. Unlike most classification models tested in specific domains, ISW-LM overcomes domain-specific limitations and applies to various situations. ISW-LM predicts the diagnosis and risk of COVID-19 acute illness according to the patient's clinical and laboratory parameters. ISW-LM has three symptom weighting functions that add different symptom weights for COVID-19 patients than regular patients.

The use of ISW-LM is for early diagnosis of suspected cases of COVID-19. Removing incomplete data in the dataset makes it possible to avoid these data from affecting the test results. ISW-LM combines fuzzy logic and data classification methods to deal with uncertain and inaccurate data. As shown in Figure 3, the process of ISW-LM processing datasets includes five stages (in the order of operation): data processing, symptoms' weight function, symptoms based on importance and weight, symptom weight, and patient attribute prediction or diagnosis.

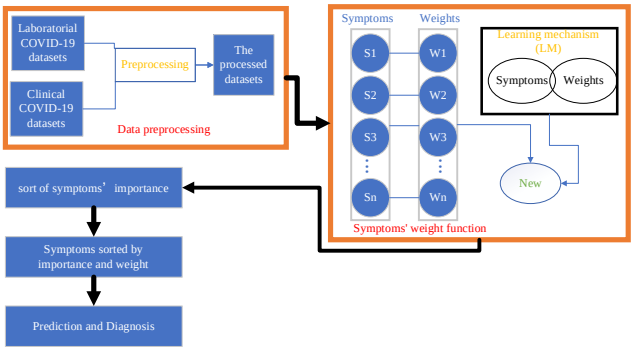

Figure 3. The flow chart of ISW-LM.

A learning mechanism (LM) and a genetic algorithm (GA) can combine. For example, the overall algorithm of deep learning and GA combination can be employed for learning and optimization [23]. The deep learning (DL) mechanism can automatically generate data-driven prediction functions for different objectives. This mechanism uses the prediction functions to optimize the objectives. Convert the above optimization objective prediction function into an alternative model, and combine it with GA to generate a dataset containing the best results [24]. DL is widely used in driverless cars, image recognition and classification, machine translation, target recognition, emotion recognition, and art creation. In general, in driverless cars, picture recognition and classification, and target recognition, the role of DL is to recognize and detect objects. The application of DL in human-like brain understanding, human emotion, and creation are reflected in machine translation, emotion recognition, and art creation, respectively. In the medical field, DL can be used to diagnose eye diseases and develop visual aids [25]. It can also improve people's lives with low vision by providing them with assisted reading functions and obstacle avoidance systems and helping them create art.

The DL mechanism has three learning modes: supervised, unsupervised, and semi-supervised. The three learning modes have significant differences in the characteristics of the training set: the training set of unsupervised learning only includes input; the training set of supervised learning includes input and expected output; the training set of semi-supervised learning includes input, but it will also provide additional information to the network during training. Hu, et al. [26] proposed a new deep-supervised learning framework for multi-scale feature learning of image classification. The expansion of the VGG-16 network resulted in this deeply-supervised network. The network has multiple convolution layers. After each convolution layer, there is batch normalization (BN). Finally, the feature map is input to the entire connection layer classification. Creating the model of this in-depth supervised learning framework can solve the data imbalance problem of COVID-19 based on CT scan diagnosis. And BN can accelerate network convergence and improve model robustness. Although ISW-LM can

predict sore throats with high accuracy, it still needs to improve its accuracy in accurately predicting key symptoms such as dyspnea and fever.

2.7. GAN

GAN's full name is Generative Adversarial Networks. As shown in Figure 4, GAN consists of a generator and a discriminator. It can already generate class-specific images [27]. GAN contains both discriminative and generative models. The generative network G is to fool the discriminative network D with the generated images. Relatively, the role of discriminative network D is to identify whether a particular photo was caused by G or not. This model training process is equal to the operation of gaming. And the capabilities of G and D in this process are gradually improved during training.

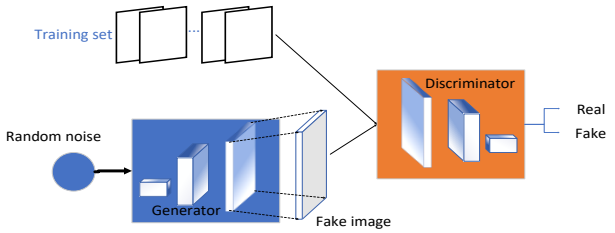

Figure 4. Structure of GAN.

If x represents the picture and z denotes the random noise that generates the image. Then $G(z)$ represents the data generated by G and input to D. The $D(x)$ output from D represents the probability that it is the real picture. And the picture input to D is real when $D(x) = 1$. On the contrary, when $D(x) = 0$, the input to D is not the real picture. The training reaches equilibrium when the ability of G improves to the point where it is no longer able to distinguish between real and fake images. At this point, if $D[G(z)]$ denotes the probability that the image $G(z)$ generated by G is recognized by D as the "real" image, then

$$D(x) = D[G(z)] = 0.5 \tag{5.}$$

Based on GAN, Gulakala, et al. [13] proposed a Progressively Growing Generative Adversarial Network (PGGAN) for bridging dataset data and enhancing data. They also proposed additional new performance-enhanced CNNs for Chest X-ray (CXR) image classification, which is beneficial for improving detection accuracy.

For the learning of D and G, the G first generates some fake images, which are output to the D. At the same time, real pictures in the dataset are also output to the discriminator. The discriminator recognizes the authenticity of the image and gives the judgment result. At the network's first run, the discriminator completes a round of learning. Since the GAN is of tandem connection, let the discriminator parameters not vary. Instead, let it only pass the error to the generative network, and the generator completes a round of learning immediately afterward. Shabani, et al. [14] proposed a dual-model framework for segmenting COVID-19 CT scan. This model framework requires no additional pixel-level annotation of the dataset. Subtract the health images generated by GAN from the original CT image to extract the picture of the 3D infection area for diagnosis. Their dual-model architecture outperforms state-of-the-art unsupervised and weakly supervised segmentation methods in COVID-19 CT image segmentation performance. However, GAN requires health data as input. Because the clinical database of the hospitals does not contain health data in their clinical databases, it will affect the performance of GAN in processing image data.

3. Dataset

Preprocessing is performed using a COVID-19 dataset D [28]. D contains two attributes: subject and number of CCT images. Table 1 shows the description of dataset D. The preprocessing process consists of five stages: grayscaling, histogram stretching (HS), margin and text crop (MTC), down-sampling (DS), and colorization. Figure 5 illustrates the dataset preprocessing stage.

The dataset images were from COVID-19 and HC. I preprocessed the randomly obtained data set. I first gray-scaled the dataset images and changed their brightness or contrast to make them easier to distinguish. After image grayscaling, assuming $D_1 = \{d_1(k)\}$, I first calculated its upper limit $d_1^U(k)$ and lower limit $d_1^L(k)$, which are given by:

$$\begin{cases} d_1^U(k) = \max\limits_x \max\limits_y d_1(x,y|k) \\ d_1^L(k) = \min\limits_x \min\limits_y d_1(x,y|k) \end{cases} \tag{6.}$$

I can define the HSed image as

$$d_2(k) = \frac{d_1(k) - d_1^L(k)}{d_1^U(k) - d_1^L(k)}, \tag{7.}$$

Table 1. A COVID-19 dataset

Dataset	No. of Subjects	No. of Images
D	142+142	320(COVID-19)+320(HC)

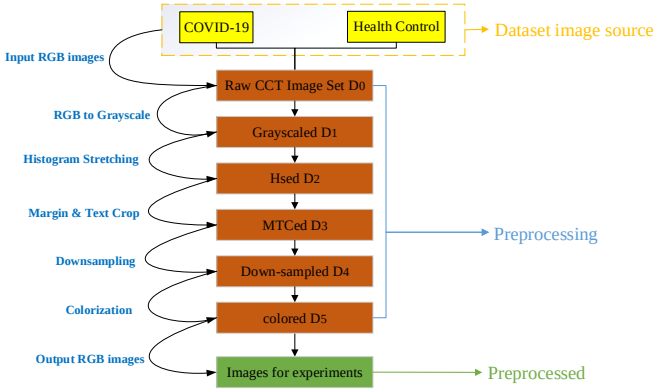

Figure 5. Preprocessing

After the histogram stretching of the grayscaled D_1, the original images with uneven distribution of gray levels expand the grayscale, and the contrast increases significantly. $D_4 = \{d(k)\}$ is the symbol of the downsampled dataset, and the dataset image size is reduced to (a_1, a_2) . The final grayscale image $d_4(k)$ is synthesized as a multichannel color image and output image $d(k)$. At this time, the size of $d(k)$ is $a_1 \times a_2 \times a_3$, and the formula for synthesizing a color image is defined as

$$d(k) = f_{cat}^{channel}[d_4(k), d_4(k), d_4(k)]. \quad (8.)$$

4. Methodology

4.1. GLCM

The gray level co-occurrence matrix is a way to describe image texture features. Its method for extracting image texture features includes four steps: gray image, gray level quantization, feature value calculation, and texture feature image generation [29, 30]. The gray level co-occurrence matrix obtains its matrix by calculating the gray level image. Then gets some eigenvalues of the matrix by calculating the co-occurrence matrix to represent some texture features of the picture. The gray level co-occurrence matrix has four angles, which are 0°, 45°, 90° and 135°.

After calculating the co-occurrence matrix, it is often not to apply directly but to calculate the texture feature quantity on this basis. Researchers often use contrast, energy, entropy, correlation, and other feature quantity to express the texture feature [31].

For the generation of the matrix, assume that there is any point (x, y) in the image, and another point $(x + d_1, y + d_2)$ that is off it. Then set the gray value of that point to (g_1, g_2). As point (x, y) moves randomly throughout the image, various (g_1, g_2) values will be obtained. If k denotes the number of levels of grayscale values, then there are k^2 combinations of (g_1, g_2). Count the number of times each type of (g_1, g_2) value appears on the whole screen and arranges into a square matrix. The angle of the gray level co-occurrence matrix depends on the case: when both (d_1) and (d_2) take different values to obtain the joint probability matrix for other cases.

When $d_1 = 1, d_2 = 0$, it is horizontal scanning;
When $d_1 = 0, d_2 = 0$, it is vertical scanning;
When $d_1 = 1, d_2 = 0$, it is 45° scanning;
When $d_1 = -1, d_2 = 0$, it is 135° scanning.

To put it simply, if the point (x, y) is adjacent to only one point with the same gray value, then the gray value is 2. If there is no point with the same gray value near the point (x, y), then the gray value is 1. Then the gray-level co-occurrence matrix is obtained. Contrast (CON), entropy (ENT), define moment (IDM), and energy (ASM) are often utilized to represent texture features.

CON measures the distribution of matrix values and the extent to which local variations in the image reflect the clarity and depth of the texture grooves. The deeper the texture groove, the greater the contrast and the clearer the effect; however, if the contrast value is small, the groove is shallow, and the effect is fuzzy.

$$con = \sum_i \sum_j (i-j)^2 p(i,j) \quad (9.)$$

ASM: The energy transformation reflects the evenness of the image's gray distribution and texture fineness. If the element values of the gray level co-occurrence matrix are similar, the energy is small, indicating that the texture is fine; if some values are large and others are small, the energy value is significant. The enormous energy value indicates a more uniform and regular texture pattern.

$$asm = \sum_i \sum_j P(i,j)^2 \quad (10.)$$

ENT: The randomness measure of the amount of information included in an image. When all the values of the co-occurrence matrix are equal, or the pixel values show the maximum randomness, the entropy is maximum; Therefore, the entropy value indicates the complexity of the gray image distribution. The larger the entropy value is, the more complex the image is.

$$ent = -\sum_i \sum_j P(i-j)\, log\, P\,(i,j) \quad (11.)$$

IDM: The deficit moment reflects the size of the local change of the image texture. If the different regions of the image texture are uniform and change slowly, the inverse variance will be more significant, and vice versa.

$$idm = \sum_i \sum_j \frac{P(i,j)}{1 + (i+j)^2} \quad (12.)$$

4.2. RVFL

RVFL is a unique single hidden layer neural network proposed by Pao, et al. [32]. Unlike deep neural networks, RVFL neural network (RVFLNN) is equivalent to directly putting hidden layers into input layers as enhancement nodes. RVFLNN performs a nonlinear transformation on the input vector at the input layer. Because RVFLNN is a simple flat network structure, RVFLNN is faster in supervised learning and training and can converge to the optimal solution within a limited number of training times. RVFL has good generalization performance [33]. RVFLNN uses fewer samples for training, and the network can achieve the required accuracy in the given dataset area. RVFL network enhances the generalization ability by using hidden layer learning to improve nonlinear kernel raw data. RVFL is a classic single-layer feedforward neural network [34]. RVFL network can randomly initialize all weights of the input layer and enhance the deviation between nodes. RVFL network has only one single-hidden layer. The following figure (Figure 6) illustrates the RVFL network architecture.

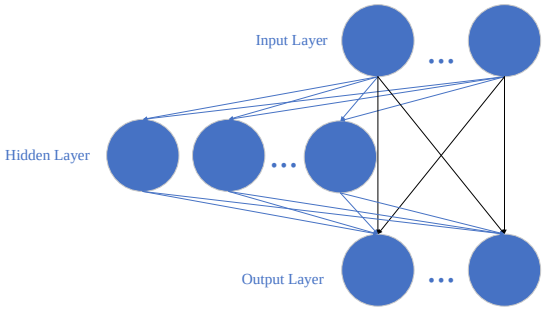

Figure 6. Structure of the RVFL network with direct links. The black line indicates the direct link from the input to the output layer.

Suppose N represents the number of input data. In that case, d represents the dimension of input data, Y represents the label corresponding to the input data, c represents the number of label types, L represents the number of nodes in the hidden layer. h represents the activation function of the hidden layer, then the formula corresponding to RVFLNN is described as follows:

$$y_i = \sum_{j=1}^{L} \beta_j h_j(x_i) + \sum_{j=L+1}^{L+d} \beta_j x_{ij}, i = 1,2,\dots,N \quad (13.)$$

The summation formula behind the above recipe represents the connected output from the input layer directly to the output layer. The following matrix shows the hidden layer output matrix.

$$\begin{bmatrix} h_1(x_1) & \cdots & h_L(x_1) & x_{11} & \cdots & x_{1d} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ h_1(x_N) & \cdots & h_L(x_N) & x_{N1} & \cdots & x_{Nd} \end{bmatrix} \quad (14.)$$

Then, the weight calculation formula of the output layer can be expressed as follows according to the least square method:

$$\beta = (H^T H)^{-1} H^T Y \quad (15.)$$

Some research results have confirmed that RVFL performs better than Extreme Learning Machine (ELM) [35].

4.3. K-fold Cross-Validation

The use of cross-validation is to estimate the performance of machine learning models for classification datasets. The data set for the cross-validation consists of training sets and test sets. The model first learns the classification on the training set [36, 37]. The test set can measure the performance of the classification. In each cross-validation, a part of the dataset becomes a test set, which will no longer become a test set in subsequent cross-validation; meanwhile, the rest is regarded as a new training set [38]. K-fold cross-validation should be implemented to test the test set with high variance [39]. K-fold cross-validation means randomly dividing the entire dataset into K equal-sized parts.

In each division, one copy is randomly selected as the test set, and the remaining $K-1$ copies are used as the training sets. When the data set is randomly divided in this way K times, then estimate the algorithm accuracy of the model by calculating the average value of the results. Because the training set is independent of the validation set, cross-validation avoids over-fitting [40], which improves the model's performance. Figure 7 shows the K-fold cross-validation process.

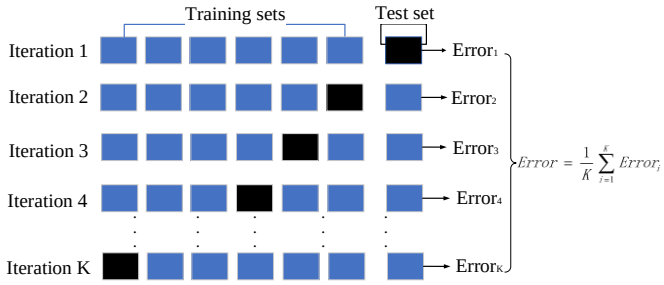

Figure 7. Structure and calculation results of K-fold cross-validation. The black and blue blocks represent the test and training sets, respectively.

4.4. Measurement

There are numerous ways to evaluate the performance of classifiers. This paper uses the confusion matrix to verify the test results. Classification results are based on confusion matrix operations [41]. There are seven

evaluation metrics of classification: sensitivity [42, 43], specificity, precision, accuracy, MCC, F1-score, and FMI [44]. To explain more clearly the formulae of the above metrics, the concepts of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) are also added. The rest of section 4 shows the relevant evaluation indicators of the above confusion matrix.

- TP: The initial sample is positive, and the predicted result is also positive, i.e., the prediction is correct.
- TN: The initial sample is negative, and the prediction result is also negative, i.e., the prediction is correct.
- FP: The initial sample is negative, but the model predicts a positive result, i.e., the prediction is wrong.
- FN: The initial sample is positive, but the model predicts a negative result, i.e., the prediction is wrong.
- Sensitivity: Sensitivity describes the proportion of all true positive classes to all classes predicted to be positive. The higher the sensitivity, the more correct the prediction.

$$Sensitivity = \frac{TP}{TP + FN} \quad (16.)$$

- Specificity: Specificity describes the proportion of identified negative classes to all negative classes. A higher specificity indicates a more negligible probability of a wrong prediction.

$$Specificity = \frac{TN}{TN + FP} \quad (17.)$$

- Precision: Precision is the proportion of correct predictions in the positive sample based on the prediction result.

$$Precision = \frac{TP}{TP + FP} \quad (18.)$$

- Accuracy: The percentage of the total number of correct predictions in positive and negative classes.

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} \quad (19.)$$

- MCC: MCC treats the true and predicted classes as two (binary) variables and calculates their correlation coefficients (similar to calculating the correlation coefficient between any two variables). The higher the correlation between the actual and predicted values, the better the prediction.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (20.)$$

- F1-score is the harmonic mean assessment index of precision and recall.

$$F1 = \frac{2TP}{2TP + FP + FN} \quad (21.)$$

- FMI: Since the clustering algorithm does not rely on the true class criteria of the samples, it cannot assess the learners' quality by calculating the misclassification (accuracy or error rate) as a classification for supervised learning as an optimization goal. There are two types of performance metrics for clustering in general: external and internal metrics.

$$FMI = \frac{TP}{\sqrt{(TP + FP)(TP + FN)}} \quad (22.)$$

5. Experiment Result and Discussions

5.1. Statistical Analysis

This study trained and tested a dataset of lung images of patients with COVID-19 and HC using a hybrid network combining RVFL and GLCM. I preprocessed the original pictures in five stages: grayscaling, histogram stretching, margin and text crop, down-sampling, and colorization [45, 46]. I synthesized multichannel images in the colorization stage. GLCM extracts the pictures of the grayscale processing stage for image features. After that, extracted feature values are output to RVFL to obtain the final classification results. The network has run ten times, and 10-fold cross-validations produced the experimental results. Table 2 shows the experimental results. As shown in Figure 8**,** the line graphs indicate the trends of model specificity, precision, and accuracy.

Table 2. Measures on ten runs.

Run	Sen	Spc	Prc	Acc	F1	MCC	FMI
1	80.62	75.94	77.01	78.28	78.78	56.62	78.80
2	77.50	75.31	75.84	76.41	76.66	52.83	76.67
3	78.12	77.50	77.64	77.81	77.88	55.63	77.88
4	77.50	75.94	76.31	76.72	76.90	53.44	76.90
5	75.94	76.56	76.42	76.25	76.18	52.50	76.18
6	78.75	76.25	76.83	77.50	77.78	55.02	77.78
7	80.31	74.06	75.59	77.19	77.88	54.48	77.91
8	81.56	71.25	73.94	76.41	77.56	53.10	77.66
9	77.81	77.19	77.33	77.50	77.57	55.00	77.57
10	80.00	73.44	75.07	76.72	77.46	53.55	77.50
Mean+SD	78.81±1.75	75.34±1.92	76.20±1.12	77.08±0.68	77.46±0.73	54.22±1.35	77.48±0.74

Table 2 displays the detailed experimental results of ten cross-validations of the hybrid network. These results include sensitivity, specificity, precision, accuracy, F1-score, MCC, and FMI. Table 2 also displays the accurate data of each test for the above items and the mean and error values obtained for the ten tests. And the results of the above items were $78.81 \pm 1.75\%$, $75.34 \pm 1.92\%$, $76.20 \pm 1.12\%$, $77.08 \pm 0.68\%$, $77.46 \pm 0.73\%$, $54.22 \pm 1.35\%$, and $77.48 \pm 0.74\%$.

From Table 2, my proposed model has a slight error. Therefore, it had better generalization ability and could effectively avoid the overfitting phenomenon. The main reason for this result is that RVFL is a random single-hidden layer neural network, and its input layer can transmit data directly to the output layer. RVFL has fewer hidden layers and nodes, which can effectively increase the generalization ability.

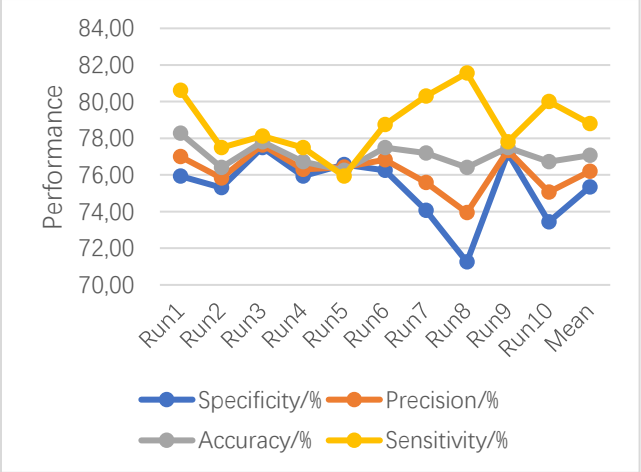

Figure 8. Trends in specificity, precision, accuracy, and sensitivity.

5.2. Comparison with State-of-the-art Approaches

Table 3. Comparison with six state-of-the-art algorithms.

Approach	Sen	Spc	Prc	Acc	F1	MCC	FMI
ELM-BA	56.91±1.21	71.94±2.17	67.01±1.52	64.42±0.88	61.53±0.77	29.19±1.88	61.74±0.77
WE+BBO	72.94±0.96	73.97±1.02	73.70±0.79	73.45±0.69	73.31±0.71	46.91±1.38	73.32±0.71
GLCM-SVM	72.38±2.68	77.38±1.96	76.22±1.21	74.88±0.86	74.21±1.25	49.85±1.70	74.25±1.21
GLCM-ELM	74.19±2.74	77.81±2.03	77.01±1.29	76.00±0.98	75.54±1.31	52.08±1.95	75.57±1.28
WE-CSO	74.75±2.02	77.19±1.41	76.63±0.89	75.97±0.80	75.66±1.04	51.97±1.56	75.68±1.02
WEGA	72.47±2.28	73.88±1.17	73.50±0.74	73.17±0.97	72.97±1.30	46.37±1.90	72.98±1.29
GLCM-RVFL (Ours)	78.81±1.75	75.34±1.92	76.20±1.12	77.08±0.68	77.46±0.73	54.22±1.35	77.48±0.74

To test the efficiency of the model I mentioned for image recognition, I used a 10-fold cross-validation method to obtain it. By comparing the performance with the six existing models ELM-BA, WEBBO, GLCM-SVM, GLCM-ELM, WE-CSO, and WEGA, from Table 3, I found that: In terms of sensitivity, my proposed model won first place with 78.81%. Regarding specificity, my model outperforms ELM-BA, WEBBO, and WEGA with 75.34% but cannot compare to GLCM-SVM with 77.38% and GLCM-ELM with 77.81%. My model is 76.20% in precision, which is lower than GLCM-SVM, GLCM-ELM, and WE-CSO, but exceeds the other three models. In terms of accuracy, my model is 77.08%, outperforming the other six models. Regarding F1-score, MCC, and FMI, my model outperforms the other six models with 77.46%, 54.22%, and 77.48%, respectively. By analyzing the above data, I can conclude that my model performs best in sensitivity, accuracy, F1 score, MCC, and FMI. Nevertheless, in terms of precision and specificity, my model was not as good as parts of the models involved in the comparison. A model that performs well in several performance evaluation metrics helps to achieve better image classification work.

6. Conclusion

This paper proposes a hybrid network model combining GLCM and RVFL. The dataset pre-processing process consists of five stages. After pre-processing, GLCM extracts CT image features. Then I outputted CT image feature data to RVFL for classification to identify which image is the category of infection with COVID-19. The preprocessing stage colors the grayscale images to form a color image by stacking grayscale images in multiple channels. Image features are input to RVFL for image classification. In the training and testing phase of the model, I used ten-fold cross-validation to evaluate the classification performance. The model was estimated from seven aspects.: Sensitivity, specificity, precision, accuracy, F1-score, MCC, and FMI, the ultimate achievement of which were $78.81 \pm 1.75\%$, $75.34 \pm 1.92\%$, $76.20 \pm 1.12\%$, $77.08 \pm 0.68\%$, $77.46 \pm 0.73\%$, $54.22 \pm 1.35\%$, and $77.48 \pm 0.74\%$. Compared with ELM-BA, WEBBO, GLCM-SVM, GLCM-ELM, WE-CSO, and WEGA, the results demonstrate that my proposed model has better sensitivity, accuracy, F1-score, MCC, and FMI. Therefore, my proposed model can better implement COVID-19 picture classification work. However, the proposed model has room for improvement in accuracy and specificity. I will continue to enhance the model in the future. I will also work to improve COVID-19 picture classification in complex environments.

References

[1] X. Xie, Z. Zhong, W. Zhao, C. Zheng, F. Wang, and J. Liu, "Chest CT for Typical 2019-nCoV Pneumonia: Relationship to Negative RT-PCR Testing," *Radiology*, vol. 296, no. 2, p. 200343, 2020.

[2] X. Yao, Z. Zhu, C. Kang, S. H. Wang, J. M. Gorriz, and Y. D. Zhang, "AdaD-FNN for Chest CT-Based COVID-19

Diagnosis," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 7, no. 1, pp. 5-14, 2023.
[3] Y. Zhang and M. A. Khan, "SNELM: squeezeNet-guided ELM for COVID-19 recognition," *Computer Systems Science and Engineering*, vol. 46, no. 1, pp. 13-26, 2023.
[4] B. Garcia Santa Cruz, M. N. Bossa, J. Solter, and A. D. Husch, "Public Covid-19 X-ray datasets and their impact on model bias - A systematic review of a significant problem," *Med Image Anal*, vol. 74, p. 102225, Dec 2021.
[5] X. Han, "A survey on deep learning in COVID-19 diagnosis," *Journal of Imaging*, vol. 9, no. 1, p. 1, 2023.
[6] S.-H. Wang and S. Fernandes, "AVNC: Attention-based VGG-style network for COVID-19 diagnosis by CBAM," *IEEE Sensors Journal*. doi: 10.1109/JSEN.2021.3062442
[7] S. Bakheet and A. Al-Hamadi, "Automatic detection of COVID-19 using pruned GLCM-Based texture features and LDCRF classification," *Comput Biol Med*, vol. 137, p. 104781, Oct 2021.
[8] R. Sharma, T. Goel, M. Tanveer, S. Dwivedi, and R. Murugan, "FAF-DRVFL: Fuzzy activation function based deep random vector functional links network for early diagnosis of Alzheimer disease," *Applied Soft Computing*, vol. 106, p. 107371, 2021/07/01/ 2021.
[9] B. B. Hazarika and D. Gupta, "Modelling and forecasting of COVID-19 spread using wavelet-coupled random vector functional link networks," *Appl Soft Comput*, vol. 96, p. 106626, Nov 2020.
[10] X. Wang *et al.*, "SSA-Net: Spatial self-attention network for COVID-19 pneumonia infection segmentation with semi-supervised few-shot learning," *Med Image Anal*, vol. 79, p. 102459, Jul 2022.
[11] L. Fang and X. Liang, "ISW-LM: An intensive symptom weight learning mechanism for early COVID-19 diagnosis," *Comput Biol Med*, vol. 146, p. 105615, Jul 2022.
[12] I. Goodfellow *et al.*, "Generative Adversarial Nets," in *Neural Information Processing Systems*, 2014.
[13] R. Gulakala, B. Markert, and M. Stoffel, "Rapid diagnosis of Covid-19 infections by a progressively growing GAN and CNN optimisation," *Comput Methods Programs Biomed*, vol. 229, p. 107262, Nov 26 2022.
[14] S. Shabani, M. Homayounfar, V. Vardhanabhuti, M. A. Nikouei Mahani, and M. Koohi-Moghadam, "Self-supervised region-aware segmentation of COVID-19 CT images using 3D GAN and contrastive learning," *Comput Biol Med*, vol. 149, p. 106033, Oct 2022.
[15] S. Lu, "A Pathological Brain Detection System based on Extreme Learning Machine Optimized by Bat Algorithm," *CNS & Neurological Disorders - Drug Targets*, vol. 16, no. 1, pp. 23-29, 2017.
[16] X. Yao, "COVID-19 Detection via Wavelet Entropy and Biogeography-based Optimization," in *COVID-19: Prediction, Decision-Making, and its Impacts*, K. C. Santosh and A. Joshi, Eds.: Springer, 2020, pp. 69-76.
[17] Y. Chen, "Covid-19 Classification Based on Gray-Level Co-occurrence Matrix and Support Vector Machine," in *COVID-19: Prediction, Decision-Making, and its Impacts*, K. C. Santosh and A. Joshi, Eds. Singapore: Springer Singapore, 2020, pp. 47-55.
[18] P. Pi, "Gray level co-occurrence matrix and extreme learning machine for Covid-19 diagnosis," *International Journal of Cognitive Computing in Engineering*, vol. 2, pp. 93-103, 2021.
[19] W. Wang, "Covid-19 Detection by Wavelet Entropy and Cat Swarm Optimization," *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, vol. 415, pp. 479-487, 2022.
[20] J.-J. Wang, "Covid-19 Detection by Wavelet Entropy and Genetic Algorithm," *Intelligent Computing Theories and Application*, vol. 13394, pp. 588-599, 2022.
[21] S. Apostolopoulos, S. D. Zanet, C. Ciller, S. Wolf, and R. Sznitman, "Pathological OCT Retinal Layer Segmentation Using Branch Residual U-Shape Networks," ed: Springer, Cham, 2017.
[22] X.-x. You, Z.-m. Liang, Y.-q. Wang, and H. Zhang, "A study on loss function against data imbalance in deep learning correction of precipitation forecasts," *Atmospheric Research*, vol. 281, 2023.
[23] H. M. Balaha, M. H. Balaha, and H. A. Ali, "Hybrid COVID-19 segmentation and recognition framework (HMB-HCF) using deep learning and genetic algorithms," *Artif Intell Med*, vol. 119, p. 102156, Sep 2021.
[24] P. Wu, Y. He, Y. Li, J. He, X. Liu, and Y. Wang, "Multi-objective optimisation of machining process parameters using deep learning-based data-driven genetic algorithm and TOPSIS," *Journal of Manufacturing Systems*, vol. 64, pp. 40-52, 2022/07/01/ 2022.
[25] J. Wang, S. Wang, and Y. Zhang, "Artificial intelligence for visually impaired," *Displays*, vol. 77, 2023.
[26] K. Hu *et al.*, "Deep supervised learning using self-adaptive auxiliary loss for COVID-19 diagnosis from imbalanced CT images," *Neurocomputing*, vol. 458, pp. 232-245, Oct 7 2021.
[27] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative Adversarial Text to Image Synthesis," in *International Conference on Machine Learning (ICML)*, 2016.
[28] Y. D. Zhang and S. Satapathy, "A seven-layer convolutional neural network for chest CT-based COVID-19 diagnosis using stochastic pooling," *IEEE Sensors Journal*, vol. 22, no. 18, pp. 17573 - 17582, 2022.
[29] H. Shayeste and B. M. Asl, "Automatic seizure detection based on Gray Level Co-occurrence Matrix of STFT imaged-EEG," *Biomedical Signal Processing and Control*, vol. 79, Jan 2023, Art no. 104109.
[30] M. Christaki, C. Vasilakos, E. E. Papadopoulou, G. Tataris, I. Siarkos, and N. Soulakellis, "Building Change Detection Based on a Gray-Level Co-Occurrence Matrix and Artificial Neural Networks," *Drones*, vol. 6, no. 12, Dec 2022, Art no. 414.
[31] A. Saihood, H. Karshenas, and A. R. N. Nilchi, "Deep fusion of gray level co-occurrence matrices for lung nodule classification," *Plos One*, vol. 17, no. 9, Sep 2022, Art no. e0274516.
[32] Y. H. Pao, G. H. Park, and D. J. Sobajic, "Learning and generalization characteristics of the random vector Functional-link net," *Neurocomputing*, vol. 6, no. 2, pp. 163-180, 1994.
[33] P.-B. Zhang and Z.-X. Yang, "A new learning paradigm for random vector functional-link network: RVFL+," *Neural Networks*, vol. 122, pp. 94-105, 2020/02/01/ 2020.
[34] W. Cui *et al.*, "Received-Signal-Strength based Indoor Positioning Using Random Vector Functional Link Network," *IEEE Transactions on Industrial Informatics*, pp. 1-1, 2017.
[35] Y. Zhang, J. Wu, Z. Cai, B. Du, and P. S. Yu, "An unsupervised parameter learning model for RVFL neural network," *Neural Networks*, vol. 112, pp. 85-97, 2019/04/01/ 2019.
[36] K. Hirose, K. Miura, and A. Koie, "Hierarchical clustered multiclass discriminant analysis via cross-validation,"

Computational Statistics & Data Analysis, vol. 178, Feb 2023, Art no. 107613.
[37] L. Du *et al.*, "Development and external validation of a machine learning-based prediction model for the cancer-related fatigue diagnostic screening in adult cancer patients: a cross-sectional study in China," *Supportive Care in Cancer*, vol. 31, no. 2, Feb 2023, Art no. 106.
[38] T. Yan, S.-L. Shen, A. Zhou, and X. Chen, "Prediction of geological characteristics from shield operational parameters by integrating grid search and K-fold cross validation into stacking classification algorithm," *Journal of Rock Mechanics and Geotechnical Engineering*, vol. 14, no. 4, pp. 1292-1303, 2022.
[39] H. L. Vu, K. T. W. Ng, A. Richter, and C. An, "Analysis of input set characteristics and variances on k-fold cross validation for a Recurrent Neural Network model on waste disposal rate estimation," *Journal of Environmental Management*, vol. 311, p. 114869, 2022/06/01/ 2022.
[40] Z. Wang *et al.*, "Enhanced RBF neural network metamodelling approach assisted by sliced splitting-based K-fold cross-validation and its application for the stiffened cylindrical shells," *Aerospace Science and Technology*, vol. 124, p. 107534, 2022/05/01/ 2022.
[41] J. Xu, Y. Zhang, and D. Miao, "Three-way confusion matrix for classification: A measure driven view," *Information Sciences*, vol. 507, pp. 772-794, 2020/01/01/ 2020.
[42] A. Gaye, N. A. Sene, P. Balland, V. Sambou, and P. B. Gning, "Extraction and physicomechanical characterisation of Typha Australis fibres: Sensitivity to a location in the plant," *Journal of Natural Fibers*, vol. 20, no. 1, Dec 2023, Art no. 2164106.
[43] A. Villanueva-Castellote, C. L. Puy, M. Cerda-Dieguez, A. Mira, and M. D. Ferrer, "Ex vivo evaluation of antibiotic sensitivity in samples from endodontic infections," *Journal of Oral Microbiology*, vol. 15, no. 1, Dec 2023, Art no. 2160536.
[44] D. Valero-Carreras, J. Alcaraz, and M. Landete, "Comparing two SVM models through different metrics based on the confusion matrix," *Computers & Operations Research*, vol. 152, 2023.
[45] X. Guo, "A Survey on Machine Learning in COVID-19 Diagnosis," *Computer Modeling in Engineering & Sciences*, vol. 130, no. 1, pp. 23-71, 2022.
[46] S.-H. Wang, K. M. Attique, and G. Vishnuvarthanan, "Deep rank-based average pooling network for COVID-19 recognition," *Computers, Materials, & Continua*, pp. 2797-2813, 2022.