Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
BLCov: A novel collaborative–competitive broad learning system for COVID-19 detection from radiology images

Guangheng Wu, Junwei Duan*
College of Information Science and Technology, Jinan University, Guangzhou, China

A R T I C L E   I N F O

Keywords:
Broad learning system (BLS)
Collaborative–competitive representation (CCR)
COVID-19 detection
CT scans
Chest X-ray images

A B S T R A C T

With the global outbreak of COVID-19, there is an urgent need to develop an effective and automated detection approach as a faster diagnostic alternative to avoid the spread of COVID-19. Recently, broad learning system (BLS) has been viewed as an alternative method of deep learning which has been applied to many areas. Nevertheless, the sparse autoencoder in classical BLS just considers the representations to reconstruct the input data but ignores the relationship among the extracted features. In this paper, inspired by the effectiveness of the collaborative–competitive representation (CCR) mechanism, a novel collaborative–competitive representation-based autoencoder (CCRAE) is first proposed, and then collaborative–competitive broad learning system (CCBLS) is proposed based on CCRAE to effectively address the issues mentioned above. Moreover, an automated CCBLS-based approach is proposed for COVID-19 detection from radiology images such as CT scans and chest X-ray images. In the proposed approach, a feature extraction module is utilized to extract features from CT scans or chest X-ray images, then we use these features for COVID-19 detection with its symptoms are often associated with the symptoms of pneumonia. And the use of image testing for COVID-19 detection is a fast and efficient method in commercial and large-scale use, which can be used to control the spread of COVID-19. The most commonly used technique to detect COVID-19 is real-time reverse transcription-polymerase chain reaction (RT-PCR) (Zu et al., 2020). Chest radiographs, such as computed tomography (CT) scans and chest X-ray images, also play an important role in the rapid diagnosis and early treatment. RT-PCR is widely recognized as a standard method of diagnosis, but there are still several disadvantages. RT-PCR is low sensitivity and requires sufficient expertise to collect viral RNA from patient’s nasopharyngeal swabs. In addition, testing by RT-PCR is time-consuming because it needs strict laboratory conditions (Panday et al., 2021). From this perspective, it is worthy to note that CT scans and chest X-ray images are alternative techniques for detecting COVID-19, which can be used along with RT-PCR (Kanne et al., 2020; Zu et al., 2020). As noted in Pan et al. (2020), COVID-19 pneumonia started showing changes in CT scans ten days after the appearance of related symptoms. In the early pandemic, CT scans and X-ray images were widely used for COVID-19 detection when under bad testing conditions. Also, there are research works suggesting that a combination of clinical image features and laboratory results may help in the early detection of COVID-19 (Shi et al., 2020; Lee et al., 2020; Li and Xia, 2020). Recent studies showed that the noticed changes in chest X-ray images and CT scans were found before the clinical features of COVID-19 appear (Chan et al., 2020). For example, Zhao et al. (2020) discovered that ground-glass opacity (GGO) or mixed GGO was found in most patients, and vascular dilation in the lesions was found in some patients. Zu et al. (2020) informed that round lung opacities were reported in chest CT scans. Hence, major discoveries can also be disclosed according to the diagnostic results of chest X-ray images and CT scans (Li and Xia, 2020).

Automatically diagnosing diseases by machine learning techniques has drawn increasing attention because it can save a large number of medical resources. Machine learning has been applied in various medical fields, such as lung segmentation (Gaál et al., 2020) and pneumonia detection using chest X-ray images (Wang et al., 2017). In recent years, more researches focus on developing AI approaches for COVID-19 detection by machine learning-based methods. Various supervised machine learning methods, such as support vector regression (SVR),
ridge regression (RIDGE), and random forest (RF) have been used for the predictions of the number of new cases (Ribeiro et al., 2020). Apart from the traditional machine learning methods, deep learning-based methods have also been applied to COVID-19 detection. For instance, Wang et al. (2020a) proposed a COVID-Net implemented on chest X-ray images, which achieved a promising accuracy in COVID-19 detection. According to different architecture and learning strategy, the COVID-Net was redesigned and a joint learning framework is further proposed based on this backbone in Wang et al. (2020b). Javaheri et al. (2021) designed a CovidCTNet to distinguish COVID-19 infection from common pneumonia and other lung diseases using CT scans. In Nasiri and Hasani (2021), they first employed DenseNet169 to extract features and then performed classification with the XGBoost algorithm. To evaluate the suitability of extreme learning machine (ELM) for COVID-19 classification, a COV-ELM classifier is proposed in Rajpal et al. (2020) using the texture and frequency features extracted from chest X-ray images. Besides, TLCoV is proposed in Das et al. (2021) using CNN, VGG-16, and ResNet-50 individually for detecting COVID-19 from chest X-ray images. CoroDet as a CNN-based model is proposed in Hussain et al. (2021) for automatic detection of COVID-19 using chest X-ray images or CT scans, which can achieve superior performances.

Deep neural networks have achieved significant outbreaks in many applications, however, deep neural networks are of complexity and there are numerous parameters to be calibrated. It is time-consuming for training a deep neural network and a large amount of computing resources need to be consumed. And existing COVID-19 automatic diagnosis models often face the problem of overfitting due to the small size of the dataset, resulting in low accuracy. Therefore, inspired by the collaborative–competitive representation (CCR) mechanism in CCRC (Yuan et al., 2018), we try to design a novel collaborative–competitive representation-based broad learning system (CCBLS) is first proposed for COVID-19 detection, and our proposed method can achieve competitive performances.

- The features are visualized by uniform manifold approximation and projection (UMAP) method. The visualization results demonstrated that the feature representations are distinguishing between the different categories, which can perform well in classification tasks.

The rest of this paper is organized as follows. In Section 2, we briefly review the classic broad learning system, sparse autoencoder, and collaborative–competitive representation mechanism. The proposed CCBLS and the details of our proposed COVID-19 detection approach are given in Section 3. In Section 4, we evaluate the performance of our proposed approach with a series of experiments and report the experimental results and analysis. Section 5 is the conclusion of this paper.

## 2. Related works

### 2.1. Broad learning system

Broad learning system (Chen and Liu, 2017) is presented as a flat and incremental learning neural network. A classic structure of BLS is shown in Fig. 1. As we can see, in BLS the original inputs are first transformed into the mapping features, then further obtaining the enhancement features from the mapping features. All these feature nodes will be connected to the ground truth matrix through the output weights. The details of BLS are introduced as follows.

Suppose the training data \( \{X, Y\} \) has \( K \) classes with \( N \) samples in total. There are \( n \) groups of nodes in the feature node, and the enhancement nodes contain \( m \) groups. The \( i \)th group of mapping feature nodes is calculated as

\[
Z_i = \zeta(XW_{a} + a_{a}), i = 1, 2, \ldots, n.
\]  
(1)

where weighted matrix \( W_{a} \) and bias term \( a_{a} \) are randomly generated, and \( \zeta(\cdot) \) is the \( i \)th feature mapping function which is usually a nonlinear function. Next, collecting all the \( n \) groups of mapping features together, i.e. \( Z^* = \{Z_1, Z_2, Z_3, \ldots, Z_n\} \). By feeding the obtained \( Z^* \) into enhancement feature nodes, the \( j \)th group of enhancement feature nodes can be expressed as

\[
H_j = \eta_j(Z^*W_{b} + \gamma_{b}), j = 1, 2, \ldots, m.
\]  
(2)

where \( \eta_j(\cdot) \) is also a nonlinear activation function. Similarly, \( W_{b} \) and \( \gamma_{b} \) are also randomly initialized. These enhancement feature nodes are denoted as \( H^* = \{H_1, H_2, H_3, \ldots, H_m\} \). Composing the mapping nodes
and enhancement nodes together, we can obtain a new broad feature $B = \{Z^n|H^n\}$. And finally, BLS can be expressed as

$$Y = BW = \{Z^n|H^n\}W.$$  

(3)

Then the key point of BLS is to get the solution of $W$ efficiently. Fortunately, by using the pseudo-inverse algorithm, the desired output $W$ can be calculated as

$$W = B^+Y.$$  

(4)

However, it may be too costly to compute $B^+$ by some common methods because of the large dimension of training data. Alternatively, we can solve this problem by minimizing the least square regression function as

$$\arg\min_W \|BW - Y\|_2^2 + \lambda\|W\|_2^2,$$  

(5)

where $\lambda$ is a regularization parameter. So $W$ can be formulated as

$$W = (\lambda I + BB^T)^{-1}B^TY.$$  

(6)

Specially, we have $B^+$ computed as

$$B^+ = \lim_{\delta \to 0} (\lambda I + BB^T)^{-1}B^T.$$  

(7)

### 2.2. Sparse autoencoder and collaborative–competitive representation mechanism

Since BLS generates the mapping features by randomly initializing the connecting weights, in order to overcome the randomness, a sparse autoencoder is adopted to fine-tune the random features and give a sparse representation. As we can see, the random features $Z$ are generated as equation $Z = XW$, where $W$ is randomly initialized. And the loss function of SAE can be defined as

$$Loss_{SAE} = \|Z\hat{W} - X\|_2^2 + \lambda\|W\|_2^2,$$  

(8)

where $\hat{W}$ is the solution of the sparse autoencoder, $\lambda > 0$ is a predefined parameter. To solve the issue of minimization of the loss function Eq. (8), several methods such as fast iterative shrinkage thresholding algorithm (FISTA) (Beck and Teboulle, 2009) and alternating direction method of multipliers (ADMM) (Boyd et al., 2011) are usually adopted. Here, the $l_1$ penalty applied in Eq. (8) can make the autoencoder get more sparse features, which is the same as hierarchical extreme learning machine (H-ELM) (Tang et al., 2015). To extract more dense features, $l_2$ penalty is usually adopted and then we can get another loss function of ELMAE (Kasun et al., 2016), which can be denoted as Eq. (9).

$$Loss_{ELMAE} = \|Z\hat{W} - X\|_2^2 + \lambda\|\hat{W}\|_2^2.$$  

(9)

It is much easier to compute $\hat{W}$ just by taking the derivative operation, and the solution is termed as

$$\hat{W} = (Z^TZ + \lambda I)^{-1}Z^TX,$$  

(10)

where the matrix $(Z^TZ + \lambda I)$ is generally nonsingular.

The collaborative–competitive representation mechanism is first introduced into CCRC model (Yuan et al., 2018), which makes CCRC model achieve superior performance to CRC model (Zhang et al., 2011). The loss function of the CCRC model is defined as

$$\min \sum_k \|y - X\beta_k\|_2^2 + \mu_1\|\beta_k\|_2^2 + \mu_2 \sum_{i=1}^{C} \|y - X\beta_i\|_2^2,$$  

(11)

where $y$ denotes the label vector, $X$ is the training data and $C$ notes the number of categories. $\mu_1$ and $\mu_2$ are two balancing parameters. $\|y - X\beta_k\|_2^2$ and $\sum_{i=1}^{C} \|y - X\beta_i\|_2^2$ denote the collaborative–competitive mechanism of data. Here, the first term aims to represent the test sample with all training samples collaboratively, while the second term promotes the competitive representation between different classes. Currently, there are several classical methods for feature representation such as sparse representation (SR) (Beck and Teboulle, 2009), collaborative representation (CR) (Zhang et al., 2011), and collaborative–competitive representation (CCR) (Yuan et al., 2018). As we know, sparse representation mainly performs sparse decomposition to obtain the coefficients for representing the features of original data according to the dictionary. When the size of the dictionary is huge, the process of sparse decomposition is usually very slow and the computational complexity of SR is also very large. Collaborative representation usually adopts all the training samples to collaboratively represent the features, nevertheless, it fails to make full use of the localities and discrimination information of data, which may degrade the representation-based classification performance. The collaborative–competitive representation not only considers the collaborative reconstruction of the features but also adds the crucial discrimination among different samples for better feature representation. Based on the advantages of collaborative–competitive representation (Gou et al., 2019, 2021; Li et al., 2020), we consider adopting this mechanism of feature representation and introducing it to our proposed model for better classification performance.

### 2.3. BLS and COVID-19 detection

BLS has been widely used in the fields of image processing and computer vision, such as facial recognition, human action recognition, and image classification. For example, Han et al. (2020) designed a Personalized BLS to recognize personalized expression by taking the emotional information entropy as mapping features. Zhang et al. (2019) introduced the concept of feature blocks to BLS for processing facial data. Dang et al. (2020) proposed a DNet to recognize human actions, which feeds the extracted spatial-temporal features into BLS. BLS has also achieved good results in other tasks, such as event-based object classification (Gao et al., 2020), Chinese herbal medicine classification (Cai et al., 2019), and robotic material recognition (Wang et al., 2019). BLS is also applied to many other fields, such as medical data analysis, system modeling, and fault detection (Gong et al., 2019; Chu et al., 2020; Liu et al., 2019). From these applications of BLS, we can see that BLS often works along with good feature representations. With good feature representation of sparse autoencoder and the advantages of random vector functional link neural network (RVFLNN), BLS can achieve better performance in classification problem (Chen and Liu, 2017).

In addition to sparse autoencoder, there are other kinds of autoencoders for better feature representation. For instance, collaborative–competitive representation (CCR) can effectively represent the features of data by collaborative–competitive mechanism, and it has been widely applied to image classification. Gou et al. (2019) propose a weighted discriminative collaborative–competitive representation-based (WDCCR) classifier, which not only considers the competitive representation of each class but also enhances the inter-class discrimination to promote the competitive representation of the ground truth. Besides, they designed a locality-constrained weighted collaborative–competitive representation-based classification (LWCCRC) (Gou et al., 2021) to make full use of the localities and discrimination information of data. Li et al. (2020) proposed a sparse and collaborative–competitive representation-based classification (SCCRC), which fuses SRC and CCR by multiplying their coefficients. The use of CCR results in a noticeable improvement in image classification tasks. Therefore, in BLS, it is reasonable to achieve better classification performance by replacing the sparse autoencoder with CCRAE. In most cases, COVID-19 detection from radiology images can be recognized as a classification problem. For achieving better performance of classification, an effective and efficient automated detection model is urgently proposed for COVID-19 detection. Fortunately, as a flat neural network, BLS can achieve outstanding performance with low time consumption in classification and regression tasks. Moreover, BLS has been applied to the forecast of the trend of COVID-19 infection. Zhan et al. (2021) proposed a hybrid model named RF-Bugging-BLS to predict COVID-19, and the
results indicated that the RF-Bagging-BLS model can get a promising performance in timely short-term forecasts. From the analysis, both the advantages of BLS and CCRAE give the potential possibility for using BLS with CCRAE for COVID-19 detection from radiology images.

3. Proposed method

In this section, we first propose a CCRAE algorithm to better generate the feature representation. Then we propose the CCBLS algorithm based on CCRAE for COVID-19 detection.

3.1. Collaborative–competitive Representation-based Autoencoder (CCRAE)

Inspired by the effectiveness of the collaborative–competitive representation (CCR) mechanism, we propose a CCR-based autoencoder to explore the relationship among features and expect a better feature representation in this paper. Assumed that the original data has a form to explore the relationship among features and expect a better feature representation based on CCRAE for COVID-19 detection.

The network structure of CCBLS: CCRAE is on the left, and the connecting weight of enhancement nodes is fine-tuned with competitive representation (CR) on the right side.

Note that the proposed CCRAE is different from ELMAE. ELMAE just utilizes the collaborative ability of all features. CCRAE not only utilizes the collaborative ability of all features but also the competitive ability of each feature. So that CCRAE can promote features to competitively reconstruct the original data. The proposed CCRAE algorithm is summarized in Algorithm 1.

Algorithm 1: Collaborative–competitive representation-based autoencoder (CCRAE)

Input: original data $X \in R^{N \times d_x}$; randomly generated feature $Z \in R^{N \times d_z}$; the predefined parameters $\lambda_1$ and $\lambda_2$.

Output: the output weights $\hat{W}^T \in R^{d_y \times k_F}$.

1. Calculate $M$ by Eq. (14);
2. Calculate $P = (1 + \lambda_2)(Z^T Z + \lambda_1 I + \lambda_2 M)^{-1} Z^T$ with $\lambda_1$ and $\lambda_2$;
3. Calculate $\hat{W}$ by Eq. (13);
4. Return the desired weights $\hat{W}^T$.

3.2. Collaborative–competitive Representation-based Broad Learning System (CCBLS)

Based on CCRAE, we proposed a novel broad neural network termed CCBLS. The structure of CCBLS is given in Fig. 2. As shown in Fig. 2, we first adopt CCRAE to generate the mapping features. The mapping features in CCBLS are rewritten as follows.

$$Z_i = \zeta_i(X \hat{W}^T), i = 1, 2, 3, \ldots, n.$$  

where $\hat{W}^T$ is obtained by CCRAE and $\zeta_i(\cdot)$ is a nonlinear function. The mapping features can be denoted as $Z^n = [Z_1, Z_2, Z_3, \ldots, Z_n]$. Similar to the classic BLS, each group of enhancement nodes is calculated as

$$H_j = \eta_j(Z^n W_{h_j}), j = 1, 2, 3, \ldots, m.$$  

where $W_{h_j}$ is generated randomly. We collect all these enhancement nodes as $H^n = \{H_1, H_2, H_3, \ldots, H_m\}$, where $H_j$ is recognized as the high-level features of $Z^n$. We cannot ensure the quality of feature representation of $H_j$ because the connecting weight $W_{h_j}$ is generated randomly. Therefore, we also adopt competitive representation mechanism to fine tune $W_{h_j}$. The transformed features are denoted as $E = H^n W_{hn} \in R^{N \times d_z}$, where $W_{hn}$ is randomly initialized, and $k_z$ is the number of transformed features. The fine-tuned weight $\hat{W}_{hn}$ can be calculated as Eq. (18).

$$\hat{W}_{hn} = \arg \min_{W_{hn}} \sum_{i=1}^{k_z} \|E_i \hat{W}_{hn} - H^n_i\|_F^2,$$  

where $E_i$ is the $i$-th element of $E$.
where \( \hat{E}_i = [0, \ldots, 0, E_i, \ldots, 0] \). The location of \( E_i \) in \( \hat{E}_i \) is the same as \( E \).

The above equation can be solved by taking the derivation operation and the solution can be written as

\[
\hat{W}^*_{H^m} = M_{H^m}^{-1}E^T H^m.
\]

(19)

where \( M_{H^m} \) has a similar form as Eq. (14)

\[
M_{H^m} = \begin{bmatrix}
E_1^T & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & E_{k_2}^T E_{k_2}
\end{bmatrix}
\]

(20)

Collecting the mapping feature nodes and the transformed enhancement feature nodes together, CCBLS could be expressed as

\[
Y = \{Z^m|H^m\hat{W}^T_{H^m}\}W.
\]

(21)

Let \( L = \{Z^m|H^m\hat{W}^T_{H^m}\} \), just as the classical BLS, the weights of CCBLS model can be computed as

\[
W = (\lambda I + LL^T)^{-1}L^TY.
\]

(22)

where \( \lambda \) is a hyperparameter. The algorithm of proposed CCBLS is summarized in Algorithm 2.

### 4. Materials and experiments

In this section, we will describe our proposed method termed BLCov, which consists of a feature extraction module and CCBLS for the COVID-19 detection task. Then we evaluate the effectiveness of the proposed BLCov with extensive experiments. Our proposed model is illustrated in Fig. 3.

#### 4.1. Datasets

The proposed approach is applied to detect COVID-19 from radiology images such as CT scans or chest X-ray images. We adopt three publicly available datasets including ChestX-ray8 (Wang et al., 2017), COVID-CT (Yang et al., 2020), and SARS-CoV-2 (Angelov and Almeida Soares, 2020) to evaluate our proposed approach, in which ChestX-ray8 is a collection of chest X-ray images, both COVID-CT and SARS-Cov-2 are CT scans. The brief descriptions of these datasets are given as follows.

**ChestX-ray8** This dataset contains 1125 chest X-ray images, including 500 images with no findings, 500 images with pneumonia, and 125 images with COVID-19. Some sample images of ChestX-ray8 are shown in Fig. 4.
Algorithm 2: Collaborative–competitive representation-based broad learning system (CCBLS)

Input: training data $X \in \mathbb{R}^{N \times k_0}$, $Y \in \mathbb{R}^{N \times k}$;
groups of mapping feature nodes $n$;
groups of enhancement feature nodes $m$;
the predefined parameters $\lambda, \lambda_1$, and $\lambda_2$.

Output: the predicted label $Y \in \mathbb{R}^{N \times k}$.

1. $i \leftarrow 1, j \leftarrow 1$;
2. while $i \leq n$ do
3. Calculate $\hat{W}_i^T$ by CCREAE with parameters $\lambda, \lambda_1$, and $\lambda_2$;
4. Calculate $Z_i$ with Eq. (16);
5. $i \leftarrow i + 1$;
6. Get the mapping features $Z^n$;
7. while $j \leq m$ do
8. Calculate $H_j$ according to Eq. (17);
9. $j \leftarrow j + 1$;
10. Get the enhancement features $H^m$;
11. Calculate $\hat{W}_H^T$ according to Eq. (18);
12. Set all features as $L = \{Z^n|H^m\hat{W}_H^T\}$;
13. Calculate $W$ according to Eq. (22);
14. Return the predicted label $Y = XW$.

**COVID-CT** This dataset consists of 397 CT scans from 171 patients without COVID-19 and 349 CT scans from 216 patients containing clinical findings of COVID-19.

**SARS-CoV-2** This dataset includes 2482 CT scans from 120 patients in total, among which 1252 CT scans are labeled as COVID-19 and the other 1230 are non COVID-19. Fig. 5 displays the sample images in COVID-CT and SARS-CoV-2 datasets.

| Table 1 |
|---------|
| A summary of the adopted datasets. |
| Datasets | Type | COVID-19 | Non COVID-19 | Patients |
| ChestX-ray | chest X-ray images | 125 | 1000 | – |
| COVID-CT | CT scans | 349 | 397 | 387 |
| SARS-CoV-2 | CT scans | 1252 | 1230 | 120 |

This size $512 \times 512$, and apply min–max normalization to them to ensure the uniformity of these images.

**4.2. Training process**

The ChestX-ray8 dataset is used in two phases. The first phase is a two-class classification problem, including classes of COVID-19 and no findings (625 images in total), the other phase is a three-class classification problem, including COVID-19, no findings, and pneumonia (1125 images in total). For the two-class classification task, we apply five-fold cross-validation to obtain the average evaluation metrics, and for the three-class problem, we randomly sampled 80% of the dataset for training, and the remaining 20% are used for testing. COVID-CT dataset is also used in two phases. In the first phase, we split the dataset into two parts: 746 CT scans for training and the other 204 for testing, which refers to the experimental setting in Yang et al. (2020). In the second phase, we conduct five-fold cross-validation on it in our experiments. For the SARS-CoV-2 dataset, we adopt cross-validation just similar to the second phase of the COVID-CT dataset.

We implemented the proposed method using NumPy and scikit-learn in Python 3.9. And the experiments were carried out on Google Colaboratory. As for the parameters of BLS and CCBLS, we adopt the grid search method to obtain the best values. $n$ is searched in the range of $[1,200]$ and $m$ is searched in the range of $[100,200]$. And in both BLS and CCBLS, $\lambda$ is fixed to $2^{-30}$, $\lambda_1$, $\lambda_2$ are chosen from the set $\{10^{-7}, 10^{-6}, \ldots, 10^{2}, 10^{3}\}$. The evaluation metrics used to assess the performance of models are: Sensitivity(%), Accuracy(%), Specificity(%), Precision(%), F1-score(%) and AUC(%).
4.3. Feature extraction

As noted in Haralick et al. (1973), texture features play an important role in the tasks of image classification. In the stage of feature extraction, two types of features: texture and frequency-based features, are taken into account. What is more, there are three groups of texture features. The first group is the first-order features which are directly extracted from the original image. The first-order features include energy, total energy, entropy, standard deviation, skewness, kurtosis, and so on. Another two groups of texture features are obtained from the gray-level co-occurrence matrix (GLCM) (Zare et al., 2013), and gray-level difference matrix (GLDM) (Kim and Park, 1999) accordingly. Apart from texture features, frequency-based features also play a significant role in the classification of medical images (Varuna Shree and Kumar, 2018). In this work, we apply discrete wavelet transform (DWT) to the images to obtain the frequency features. Finally, we concatenate the texture feature vector of length 54 and the frequency feature vector of length 216 to get a feature vector of size 270 for each image.

4.4. Experimental results

For the ChestX-ray8 dataset, the average test accuracy of the two-class problem is 99.68% and the accuracy of the three-class problem is 91.56%. What is more, the confusion matrices of each fold in two-class problem are shown in Fig. 6, the confusion matrix and the receiver operating characteristic curve (ROC) for three-class problem is shown in Fig. 7. A comparison of the proposed approach with the other previous methods for the three-class classification problem can be seen.
In Table 3, where DarkCovidNet (Chen et al., 2020) and DenseNet169+XGBoost (Nasiri and Hasani, 2021) are two deep learning-based methods used in the detection of COVID-19, which directly take the images as inputs. SRC (Beck and Teboulle, 2009), CRC (Zhang et al., 2011), and CCRC (Yuan et al., 2018) are three classical representation-based classification methods, which also use the similar feature representation method as the collaborative-competitive mechanism. As a classical machine learning method, SVM (Hearst et al., 1998) is also used as a compared method. What is more, the above methods take the extracted features as input data which is similar to BLS and CCBLS. As shown in Table 3, CCBLS achieves 1% improvement in Sensitivity, 1.24% in Precision, and 1.34% in Accuracy compared to BLS. Table 2 summarizes the compared methods in our experiments.

As for the COVID-CT dataset, Tables 4 and 5 show the results of the two phases respectively. In Table 4, CCBLS achieves 1.02% improvement in Sensitivity, 1.07% in Precision and 0.99% in Accuracy compared to BLS. And in Table 5, we can see that the results of CCBLS are higher than BLS on average and lower on standard deviation.

Table 6 shows the results on the SARS-CoV-2 dataset. COVIDNet (Wang et al., 2020a) is a classic deep learning-based framework for COVID-19 detection using chest X-ray images, which has achieved superior performance over other networks pretrained on ImageNet. The redesigned COVID-Net was redesigned based on COVID-Net in Wang et al. (2020b). From Table 6 we can see that both BLS and CCBLS achieve great performance, and CCBLS behaves more stably.
Table 2
A brief description of the compared methods.

| Methods          | Category       | Characteristic                                      |
|------------------|----------------|-----------------------------------------------------|
| DarkCovidNet     | Deep learning-based | X-ray images; Binary accuracy: 98.08%, multiclass accuracy: 87.02% |
| DenseNet169+XGBoost | Deep learning-based | X-ray images; Binary accuracy: 98.23%, multiclass accuracy: 89.70% |
| DenseNet169      | Deep learning-based | CT scans; Accuracy: 79.5%, F1-score: 76.0%, AUC: 90.1% |
| ResNet-50        | Deep learning-based | CT scans; Accuracy: 77.4%, F1-score: 74.6%, AUC: 86.4% |
| COVID-Net        | Deep learning-based | X-ray images; Sensitivity: 91.0%, PPV: 98.9% |
| Redesigned COVID-Net | Deep learning-based | CT scans; Accuracy: 77.07%, Sensitivity: 74.69%, AUC: 84.13% |
| SRC              | Representation-based | Extracted features                           |
| CRC              | Representation-based | Extracted features; sparse representation     |
| CCRC             | Representation-based | Extracted features; collaborative–competitive representation |
| SVM              | Machine learning | Extracted features; classical machine learning method |
| BLS              | Broad learning system | Extracted features; classical broad learning system |

Table 3
A comparison of the proposed CCBLS with other methods on the ChestX-ray8 dataset (three-class problem).

| Methods                        | Sensitivity | Specificity | Precision | F1-score | Accuracy |
|--------------------------------|-------------|-------------|-----------|----------|----------|
| DarkCovidNet (Chen et al., 2020) | 85.35       | 92.18       | 89.96     | 87.37    | 87.02    |
| DenseNet169+XGBoost (Nasiri and Hasani, 2021) | 90.05       | 93.77       | 92.59     | 91.21    | 89.70    |
| SRC                            | 68.94       | 83.03       | 83.00     | 71.15    | 70.67    |
| CRC                            | 85.45       | 90.25       | 86.93     | 86.13    | 84.00    |
| CCRC                           | 85.31       | 88.81       | 86.80     | 86.02    | 81.78    |
| SVM                            | 90.33       | 93.07       | 91.74     | 90.80    | 88.44    |
| BLS                            | 92.67       | 94.13       | 93.05     | 92.64    | 90.22    |
| CCBLS (Proposed)               | 93.67       | 94.93       | 94.29     | 93.63    | 91.56    |

Table 4
Results of the first phase of the COVID-CT dataset.

| Methods                        | Sensitivity | Precision | F1-score | Accuracy |
|--------------------------------|-------------|-----------|----------|----------|
| DenseNet-169 (Yang et al., 2020) | –           | –         | 76.00    | 79.50    |
| ResNet-50 (Yang et al., 2020)   | –           | –         | 74.60    | 77.4     |
| SRC                            | 77.55       | 62.30     | 69.09    | 66.50    |
| CRC                            | 79.59       | 68.42     | 73.58    | 72.41    |
| CCRC                           | 82.65       | 72.32     | 77.14    | 76.35    |
| SVM                            | 80.61       | 84.95     | 82.72    | 83.74    |
| BLS                            | 81.63       | 86.02     | 83.77    | 84.73    |
| CCBLS (Proposed)               | 93.67       | 94.93     | 94.29    | 93.63    | 91.56    |

Table 5
Results of the second phase of the COVID-CT dataset (mean ± sd).

| Methods                        | Sensitivity | Specificity | Precision | F1-score | Accuracy |
|--------------------------------|-------------|-------------|-----------|----------|----------|
| COVID-Net (Wang et al., 2020a)  | 57.73 ± 2.94| 64.03 ± 3.91| 61.09 ± 1.28| 63.12 ± 2.09|
| Redesigned COVID-Net (Wang et al., 2020b) | 74.69 ± 3.91| 73.48 ± 0.96| 77.04 ± 2.17| 77.07 ± 1.92|
| SRC                            | 76.82 ± 5.37| 66.36 ± 7.30| 71.13 ± 6.29| 70.63 ± 7.04|
| CRC                            | 85.72 ± 3.91| 83.38 ± 7.63| 84.25 ± 3.85| 85.00 ± 3.67|
| CCRC                           | 86.52 ± 1.74| 86.23 ± 3.79| 86.30 ± 1.57| 87.13 ± 1.64|
| SVM                            | 87.69 ± 3.99| 89.30 ± 2.39| 88.41 ± 2.04| 89.26 ± 1.75|
| BLS                            | 87.97 ± 1.69| 89.82 ± 1.78| 88.86 ± 0.80| 89.68 ± 0.77|
| CCBLS (Proposed)               | 98.00 ± 1.12| 97.93 ± 0.84| 97.96 ± 0.69| 97.94 ± 0.69|

Table 6
Results of five-cross-validation on the SARS-CoV-2 dataset (mean ± sd).

| Methods                        | Sensitivity | Specificity | Precision | F1-score | Accuracy |
|--------------------------------|-------------|-------------|-----------|----------|----------|
| COVID-Net (Wang et al., 2020a)  | 70.97 ± 2.37| 66.04 ± 2.87| 76.03 ± 1.33| 77.12 ± 0.98|
| Redesigned COVID-Net (Wang et al., 2020b) | 83.78 ± 0.62| 94.58 ± 2.07| 89.97 ± 0.91| 89.09 ± 1.08|
| SRC                            | 74.37 ± 1.03| 70.11 ± 2.33| 72.15 ± 1.24| 71.06 ± 0.83|
| CRC                            | 93.76 ± 2.05| 96.56 ± 1.24| 95.12 ± 0.81| 95.17 ± 0.66|
| CCRC                           | 93.78 ± 1.93| 96.64 ± 1.19| 95.17 ± 0.56| 95.20 ± 0.48|
| BLS                            | 97.52 ± 1.27| 98.39 ± 0.39| 97.95 ± 0.75| 97.94 ± 0.74|
| CCBLS (Proposed)               | 98.00 ± 1.12| 97.93 ± 0.84| 97.96 ± 0.69| 97.94 ± 0.69|

Table 7
Class-wise sensitivity, precision and F1-score of the ChestX-ray8 dataset (three-class problem).

| Class       | Sensitivity | Precision | F1-score |
|-------------|-------------|-----------|----------|
| COVID-19    | 100.0       | 100.0     | 100.0    |
| No findings | 98.00       | 85.22     | 91.16    |
| Pneumonia   | 83.00       | 97.65     | 89.73    |

Table 8
Class-wise sensitivity, precision and F1-score of the first phase of the COVID-CT dataset.

| Class       | Sensitivity | Precision | F1-score |
|-------------|-------------|-----------|----------|
| COVID-19    | 81.63       | 86.02     | 83.77    |
| Non COVID-19| 87.62       | 83.64     | 85.59    |
To obtain the overall performance of our proposed method, we have also computed the average class-wise sensitivity, precision, and F1-score as shown in Tables 7, 8, and 9. We also do the evaluation based on time and resource efficiency compared to a baseline BLS as well as deep learning methods on the COVID-CT dataset. The results are demonstrated in Table 10. From Table 10 we can see that the proposed CCBLS has a faster speed and cost less computational resources for it does not need to train any DNN. What is more, from Fig. 8 and Table 10 we can see that the proposed CCBLS has a faster speed than the classical BLS.

### 4.5. Visualization

In order to evaluate the effectiveness of CCBLS further, we utilize the uniform manifold approximation and projection (UMAP) (McInnes et al., 2018) method to visualize the features in the whole process of the classification. UMAP is a novel dimension reduction method based on manifold learning, which has strong mathematical foundations. The algorithm implementing this technique is competitive with the state-of-the-art dimension reduction technique t-SNE for visualization quality. What is more, UMAP can preserve the global structure and run faster. UMAP has been widely used in the fields of biological information, materials science, and machine learning.

We train the BLS and CCBLS model on the SARS-CoV-2 dataset, and we also train a BLS model termed ELMAE-BLS using ELMAE. Then we perform dimensionality reduction on the original extracted features and the processed features obtained from BLS and CCBLS with UMAP. The visualization results are illustrated in Fig. 9. In Fig. 9(a), although the original features are in a mixed structure, we can see the blurry clusters, which show the effectiveness of the extracted features. From Fig. 9(b), after being processed in the standard BLS model, the original features become more separable, but there are still many points mixing with the opposite class. As shown in Fig. 9(c) and (d), the data from the same class transformed by CCBLS and ELMAE-BLS both show distinct clustering, and CCBLS produces a better representation of the original input data, which yields a better performance.

### 5. Conclusion

In this paper, a collaborative–competitive broad learning system (CCBLS) based approach is proposed for COVID-19 detection from radiology images such as CT scans or chest X-ray images. In the proposed approach, the features are first extracted from CT scans or chest X-ray images. As an enhanced model of BLS, there are three parts in the learning strategy of CCBLS. In the first part, a collaborative–competitive representation-based autoencoder (CCRAE) is proposed to refine the mapping features. Then in the second part, we obtain the enhancement nodes based on the competitive mechanism after getting the mapping features in the first part. In the last part, the transformed features are linked to the ground truths with the output weight. The ChestX-ray8 dataset, COVID-CT dataset, and SARS-CoV-2 dataset are utilized for the experimentation. And the experimental results indicate that the accuracy of our proposed CCBLS can achieve 99.69%, 89.68%, and 97.94% for Chest X-ray8 dataset, COVID-19 CT dataset, and the SARS-CoV-2 CT dataset respectively in detecting COVID-19 and non COVID-19. Besides, our proposed approach can produce an accuracy of 91.56% for Chest X-ray8 dataset in detecting COVID-19, pneumonia, and no findings. Although CCBLS has achieved promising performance in a preliminary study with COVID-19 data, there still exist some limitations in our proposed method. The feature extraction module can be better designed in future work with the professional domain knowledge from experts. It still takes time to help doctors and radiologists in robust COVID-19 detection and assist them to treat severe cases. In future works, a modified feature extraction module will be designed to further explore the effective features in COVID-19 detection from radiology images. Additionally, our proposed method will be extended to more image data with different environmental conditions to validate its generalized performance.

### CRediT authorship contribution statement

Guangheng Wu: Conceptualization, Methodology, Validation, Visualization, Writing – original draft. Junwei Duan: Conceptualization, Methodology, Validation, Writing – review & editing, Supervision, Project administration, Funding acquisition.

### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

### Data availability

The data that has been used is confidential.

### Acknowledgments

The authors would like to thank Prof. C. L. Philip Chen and Prof. Jing Wang for their valuable suggestions to polish the manuscript. This work is supported in part by Guangdong Basic and Applied Basic Research Foundation under Grant No. 2021A1515011999, in part by the National Key Research and Development Program of China under Grant No. 2018YFC2002500, and in part by Guangdong Provincial Key Laboratory of Traditional Chinese Medicine Informatization under Grant No. 2021B1212040007.

### References

Angelov, P., Almeida Soares, E., 2020. SARS-CoV-2 CT-scan dataset: A large dataset of real patients CT scans for SARS-CoV-2 identification. MedRxiv.

Beck, A., Teboulle, M., 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. 2 (1), 183–202.

Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., et al., 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends® Mach. Learn. 3 (1), 1–122.
