Fast automated detection of COVID-19 from medical images using convolutional neural networks

Shuang Liang
School of Automation and Electrical Engineering, University of Science and Technology Beijing
Huixiang Liu
School of Automation and Electrical Engineering, University of Science and Technology Beijing
Yu Gu (guyu@mail.buct.edu.cn)
Beijing Advanced Innovation Center for Soft Matter Science and Engineering, Beijing University of Chemical Technology; School of AutoMation, Guangdong University of Petrochemical Technology; Institute of Inorganic and Analytical Chemistry, Goethe-University
Xiuhua Guo
Department of Epidemiology and Health Statistics, School of Public Health, Capital Medical University; Beijing Municipal Key Laboratory of Clinical Epidemiology, Capital Medical University
Hongjun Li
Beijing Youan Hospital, Capital Medical University
Li Li
Beijing Youan Hospital, Capital Medical University
Zhiyuan Wu
Department of Epidemiology and Health Statistics, School of Public Health, Capital Medical University; Beijing Municipal Key Laboratory of Clinical Epidemiology, Capital Medical University
Mengyang Liu
Department of Epidemiology and Health Statistics, School of Public Health, Capital Medical University; Beijing Municipal Key Laboratory of Clinical Epidemiology, Capital Medical University
Lixin Tao
Department of Epidemiology and Health Statistics, School of Public Health, Capital Medical University; Beijing Municipal Key Laboratory of Clinical Epidemiology, Capital Medical University

---

Research Article

Keywords: COVID-19, deep learning framework, convolutional neural network, X-ray, computed tomography

Posted Date: June 2nd, 2020

DOI: https://doi.org/10.21203/rs.3.rs-32957/v1

Version of Record: A version of this preprint was published on January 4th, 2021. See the published version at https://doi.org/10.1038/s42003-020-01535-7.

Abstract

Coronavirus Disease 2019 (COVID-19) is a global pandemic that poses significant health risks. The sensitivity of diagnostic tests for COVID-19 is low due to irregularities in the handling of the specimens. We propose a deep learning framework that identifies COVID-19 from medical images as an effective auxiliary testing method to improve diagnostic sensitivity. We use pseudo-coloring methods and a platform for annotating X-ray and computed tomography (CT) images to train and evaluate the convolutional neural network (CNN). The CNN achieves a performance similar to that of experts and provides high scores for multiple statistical indices, with F1 scores above 96% and specificity over 99%. Heatmaps are used to visualize the salient features extracted by the CNN. The CNN-based regression provides strong correlations between the lesion areas in the images and five clinical indicators, improving the interpretation accuracy of the classification framework. The proposed method represents a potential computer-aided diagnosis method for COVID-19 in clinical practice.

Introduction

Coronavirus Disease 2019 (COVID−19), a highly infectious disease with the basic reproductive number (R_0) of 5.7 (reported by the US Centers for Disease Control and Prevention), is caused by the most recently discovered coronavirus1 and was declared a global pandemic by the World Health Organization (WHO) on March 11, 20202. It poses a serious threat to human health worldwide, as well as significant economic losses to all countries. As of 5 May 2020, 3,525,116 people have been infected by COVID−19, and 243,540 deaths have occurred, according to the statistics of the WHO3. The Wall Street banks have estimated that the COVID−19 pandemic may cause losses of $5.5 trillion to the global economy over the next two years4. The WHO recommends using real- time reverse transcriptase−polymerase chain reaction (rRT-PCR) for laboratory confirmation of the COVID−19 virus in respiratory specimens obtained by the preferred method of nasopharyngeal swabs5. Laboratories performing diagnostic testing for COVID−19 should strictly comply with the WHO biosafety guidance for COVID−196. It is also necessary to follow the standard operating procedures (SOPs) for specimen collection, storage, packaging, and transport because the specimens should be regarded as potentially infectious, and the testing process can only be performed in a Biosafety Level 3(BSL−3) laboratory7. Not all cities in the world have adequate medical facilities to follow the WHO biosafety guidelines. According to an early report (Feb 17, 2020), the sensitivity of tests for the detection of COVID−19 using rRT-PCR analysis of nasopharyngeal swab specimens is around 30−60% due to irregularities during the collection and transportation of COVID−19 specimens8. Recent studies reported a higher sensitivity range from 71% (Feb 19, 2020) to 91% (Mar 27, 2020)9, 10. Yang et al.8 discovered that although no viral ribonucleic acid (RNA) was detected by rRT-PCR in the first three or all nasopharyngeal swab specimens in mild cases, the patient was eventually diagnosed with COVID−19 (Feb 17, 2020). Therefore, the WHO has stated that one or more negative results do not rule out the possibility of COVID−19 infection11. Additional auxiliary tests with relatively higher sensitivity to COVID−19 are urgently required.

The clinical symptoms associated with COVID–19 include fever, dry cough, dyspnea, and pneumonia, as described in the guideline released by the WHO12. It has been recommended to use the WHO's case definition for influenza-like illness (ILI) and severe acute respiratory infection (SARI) for monitoring COVID–1912. As reported by the CHINA-WHO COVID–19 joint investigation group (February 28, 2020)13, autopsies showed the presence of lung infection in COVID–19 victims. Therefore, medical imaging of the lungs might be a suitable auxiliary diagnostic testing method for COVID–19 since it uses available medical technology and clinical examinations. Chest X-ray and chest computed tomography (CT) are the most common medical imaging examinations for lungs and are available in most hospitals worldwide14. Different tissues of the body absorb X-rays to different degrees15, resulting in gray-scale images that allow for the detection of anomalies based on the contrast in the images. CT differs from normal X-ray imaging in that it utilizes X-ray beams to scan the human body to obtain information16. The CT images are digitally processed17 to create a three-dimensional image of the body. However, CT examinations are more expensive than X-ray examinations18. Recent studies reported that the use of chest X-rays and CT images resulted in improved diagnostic sensitivity for the detection of COVID–1919, 20. The interpretation of medical images is time-consuming, labor-intensive, and often subjective. The medical images are first annotated by experts to generate a report of the radiography findings. Subsequently, the radiography findings are analyzed, and clinical factors are considered to obtain a diagnosis14. However, during the current pandemic, experts are faced with a massive workload and lack of time, resulting in low diagnostic accuracy and adverse effects on the physical and mental health of the experts. Since modern hospitals have advanced digital imaging technology, medical image processing methods may have the potential for fast and accurate diagnosis of COVID–19 to reduce the burden on the experts.

Deep learning (DL) methods, especially convolutional neural networks (CNNs), are effective approaches for representation learning using multilayer neural networks21 and have provided best performance solutions to many problems in image classification22, 23, object detection24, games and decisions25, and natural language processing26. A deep residual network27 is a type of CNN architecture that uses the strategy of skip connections to avoid degradation of models. However, the applications of DL for clinical diagnoses remains limited due to the lack of interpretability of the DL model and the multimodal properties of clinical data. Some studies have demonstrated excellent performance of DL methods for the detection of lung cancer with CT images28, pneum nia with X-ray images29, and diabetic retinopathy with retinal fundus photographs30. To the best of our knowledge, the DL method has been validated only on single modal data, and no correlation analysis with clinical indicators was performed. Traditional machine learning methods are more constrained and better suited than DL methods to specific, practical computing tasks using features31. We designed a general end-to-end DL framework for information extraction from X-ray images (X-data) and CT images (CT-data) that can be considered a cross-domain transfer learning model.

In this study, we developed a custom platform for rapid expert annotation and proposed the modular CNN-based multi-stage framework (classification framework and regression framework) consisting of basic component units and special component units. The framework represents an auxiliary examination

method for high-precision and automated detection of COVID–19. This study makes the following contributions:

1. A multi-stage CNN-based classification framework consisting of two basic units (ResBlock-A and ResBlock-B) and a special unit (control gate block) was established for use with multi-modal images (X-data and CT-data). The classifiation results were compared with the experts of different levels as evaluations. Different optimization goals were established for the different stages in the framework to obtain good performances, which were evaluated using multiple statistical indicators.
2. Principal component analysis (PCA) was used to determine the characteristics of the X-data and CT-data of different categories (normal, COVID–19, and influenza). Gradient-weighted class activation mapping (Grad-CAM) was used to visualize the salient features in the images and extract the lesion areas associated with COVID–19.
3. Data preprocessing methods, including pseudo-coloring and dimension normalization, were developed to facilitate the interpretability of the medical images and adapt the proposed framework to the multi-modal images (X-data and CT-data).
4. A knowledge distillation method was adopted as a training strategy to obtain high-performance with low computational requirements and improve the usability of the method.
5. The CNN-based regression framework was used to describe the relationships between the radiography findings and the clinical symptoms of the patients. Multiple evaluation indicators were used to assess the correlations between the radiography findings and the clinical indicators.

Results

A platform was developed for annotating lesion areas of COVID–19 in medical images (X- data, CT-data). Medical imaging uses images of the interior of human bodies to create visual representations that are used for clinical diagnoses and treatment plans32. Medical images (e.g., X- data and CT-data) are acquired using digital medical imaging techniques and are typically stored in the Digital Imaging and Communications in Medicine (DICOM) format33. X-data are two-dimensional grayscale images, and CT-data are three-dimensional data, consisting of slices of the data in the z-axis direction of a two-dimensional grayscale image. Machine learning methods are playing increasingly important roles in medical image analysis, especially DL methods. DL uses multiple non-linear transformations to create a mapping relationship between the input data and output labels34. The objective of this study was to annotate lesion areas in medical images with high accuracy. Therefore, we developed a pseudo-coloring method to convert the original grayscale images to color images using the open-source image processing tools Open Source Computer Vision Library (OpenCV) and Pillow. Examples of the pseudo-color images are shown in Fig. 1.

We developed a platform that uses a client-server architecture to annotate the potential lesion areas of COVID–19 on the radiography images. The platform can be deployed on a private cloud for security and local sharing. X-data from 212 patients diagnosed with COVID–19 were analyzed by two experts to

determine the lesion areas. The X-data were collected from the public covid-chestxray-dataset35, and the images were resized to 512 x 512. Each image contained 1- 2 suspected areas with inflammatory lesions (SAs). CT-data from 95 patients diagnosed with COVID–19 and 50 patients diagnosed with influenza were annotated by the two experts using a rapid keystroke-entry format. The images of the CT scans were collected using the PHILIPS Brilliance iCT 256 system. The slice thickness of the CT scans was 5 mm, and the CT-data images were grayscale images with 512 x 512 pixels. Areas with 2–5 SAs were annotated in the images for each case, and these areas ranged from 16 x 16 to 64 x 64 pixels. Five clinical indicators (white blood cell count, neutrophil percentage, lymphocyte percentage, procalcitonin, C-reactive protein) were also obtained, as shown in Supplementary Table 1. As a control, we randomly selected 5,000 normal cases from a public dataset (Kaggle RSNA)36. The X-data of the normal cases (XNDS) and that of the COVID–19 cases (XCDS) constituted the X dataset (XDS). We collected additional CT-data of 120 cases from a public lung CT dataset (LUNA–16, a large dataset for automatic nodule detection in the lungs)37. It was confirmed by the two experienced radiologists that no lesion areas of COVID–19 or influenza were present. The CT-data of the COVID–19 cases (CTCDS), the influenza cases (CTIDS), and the normal cases (CTNDS) constituted the clinically-diagnosed CT dataset (CTDS). The images of the SAs and the clinical indicator data constituted the correlation analysis dataset (CADS). We split the XDS, CTDS, and CADS into the training-validation (trainval) part and test part. The details of the three datasets are shown in Table 1. The train-val part of CTDS is referred to as CTTS, and the test part is called CTVS. The same naming scheme was adopted for XDS and CADS, i.e., XTS, XVS, CATS, and CAVS, respectively.

PCA was used to determine the characteristics of the medical images for the COVID–19, influenza, and normal cases. PCA was used to visually compare the characteristics of the medical images (X-data, CT-data) for the COVID–19 cases with those of the normal and influenza cases, including the XNDS, the XCDS, the CTCDS, the CTIDS, and the CTNDS. Figure 2 shows the mean image of each sub-dataset and the five eigenvectors that represent the principal components of PCA in the corresponding feature space. Significant differences are observed between the COVID–19, influenza, and normal cases, indicating the possibility of being able to distinguish COVID–19 cases from normal and influenza cases.

The CNN-based classification framework exhibited excellent performance based on the validation by experts using multi-modal data. The structure of the proposed framework, consisting of the stage I sub-framework and the stage II sub-framework, is shown in Fig. 5-a, where Q, L, M, and N are the hyper-parameters of the framework for general use cases. The values of Q, L, M, and N were 1, 1, 2, and 2, respectively, in this study; this framework referred to as the CNNCF framework. The stage I and stage II sub-frameworks were designed to extract features corresponding to different optimization goals in the analysis of the medical images. The performance of the CNNCF was evaluated using multimodal datasets (X-data and CT-data) to ensure the generalization and transferability of the model, and several evaluation indicators were used (sensitivity, precision, specificity, F1, kappa). The salient features of the images extracted by the CNNCF were visualized in a heatmap (four examples are shown in Supplementary Fig. 1). In this study, four experiments were conducted; the following results were obtained.

1. Experiment-A. The results of the five evaluation indicators for the comparison of the COVID- 19 cases and normal cases for the XVS are shown in Table 2. Three experts evaluated the images, i.e., a 7th-year respiratory resident (Respira.), a 3rd-year emergency resident (Emerg.), and a 1st-year respiratory intern (Intern). An excellent performance was obtained, with the best score of F1 of 96.72%, a kappa of 95.40%, a specificity of 99.33%, and a precision of 98.33%. The sensitivity index was 95.16%, which was higher than that of the Intern (93.55%) and lower than that of the Respire. (100%) and Emerg. (100%). The receiver operating characteristic (ROC) scores for the CNNCF and the experts are plotted in Fig. 3-a; the area under the ROC curve (AUROC) of the CNNCF is 0.9961. The precision-recall scores for the CNNCF and the experts are plotted in Fig. 3-d; the area under the precision-recall curve (AUPRC) of the CNNCF is 0.9910.
2. Experiment-B. The results of the five evaluation indicators for the CTVS are shown in Table 3. The CNNCF achieved the highest performance and the best score of all five evaluation indices. The ROC scores are plotted in Fig. 3-b; the AUROC of the CNNCF is 1.0. The precision-recall scores are shown in Fig. 3-e, and the AUPRC of the CNNCF is 1.0.
3. Experiment-C. The results of the five evaluation indicators for the CTVS are shown in Table 3. The CNNCF exhibits good performance for the five evaluation indices, which are similar to that of the Respire. and higher than that of the Intern and the Emerg. The ROC scores are plotted in Fig. 3-c; the AUROC of the CNNCF is 1.0. The precision-recall scores are shown in Fig. 3-f; the AUPRC of the CNNCF is 1.0.
4. Experiment-D. The boxplots of the five evaluation indicators, the kappa coefficient, and the specificity are shown in Fig. 4, and the precision and sensitivity are shown in Supplementary Fig. 4. A bootstrapping method38 was used to calculate the empirical distributions, and McNemar's test39 was used to analyze the differences between the CNNCF and the experts. The p-values of the McNemar's test (Supplementary Table 2–4) for the five evaluation indicators were all 1.0, indicating that there was no statistically significant difference between the CNNCF results and the expert evaluations.

Introspection studies identify salient features of COVID–19. In clinical practice, the diagnostic decision of a clinician relies on the identification of the SAs in the medical images by radiologists. The statistical results show that the performance of the CNNCF for the identification of COVID–19 is as good as that of the experts. A comparison consisting of two parts was performed to evaluate the discriminatory ability of the CNNCF. In the first part, we used Grad-CAM, which is a non-intrusive method to extract the salient features in medical images, to create a heatmap of the CNNCF result. Supplementary Fig. 1 shows the heatmaps of four examples of COVID–19 cases in the XDS and CTDS. In the second part, we used density-based spatial clustering of applications with noise (DBSCAN) to calculate the center pixel coordinates (CPC) of the salient features corresponding to COVID–19. All CPCs were normalized to a range of 0 to 1. Subsequently, we used a significance test (ST)40 to analyze the relationship between the CPC of the CNNCF output and the CPC annotated by the experts. A good performance was obtained, with a mean square error (MSE) of 0.0108, a mean absolute error (MAE) of 0.0722, a root mean squared error

(RMSE) of 0.1040, a correlation coefficient (r) of 0.9761, and a coefficient of determination (R2) of 0.8801.

A strong correlation was observed between the lesion areas detected by the proposed framework and the clinical indicators. In clinical practice, multiple clinical indicators are analyzed to determine whether further examinations (i.e., medical image examination) are needed. These indicators can be used to assess the predictive ability of the model. In addition, various examinations are required to perform an accurate diagnosis in clinical practice. However, the correlations between the results of various examinations are often not clear. We used the stage II sub-framework and the regressor block of the CNNRF to conduct a correlation analysis between the lesion areas detected by the framework and five clinical indicators (white blood cell count, neutrophil percentage, lymphocyte percentage, procalcitonin, C-reactive protein) of COVID–19 using the CADS. The inputs of the CNNRF were the lesion area images of each case, and the output was a 5-dimensional vector describing the correlation between the lesion areas and the five clinical indicators.

The MAE, MSE, RMSE, r, and R2 were used to evaluate the results. The ST and the Pearson correlation coefficient (PCC)41 were used to determine the correlation between the lesion areas and the clinical indicators. A strong correlation was obtained, with MSE = 0.0163, MAE = 0.0941, RMSE = 0.1172, r = 0.8274, and R2 = 0.6465. At a significance level of 0.001, the value of r was 1.27 times the critical value of 0.6524. This result indicates a high and significant correlation between the lesion areas and the clinical indicators. The PCC was 0.8274 (range of 0.8–1.0), indicating a strong correlation. The CNNRF was trained on the CATS and evaluated using the CAVS. The initial learning rate was 0.01, and the optimization function used was the stochastic gradient descent (SGD) method42. The parameters of the CNNRF were initialized using the Xavier initialization method43.

Discussion

We developed a computer-aided diagnosis method for the identification of COVID–19 in medical images using DL techniques. Strong correlations were obtained between the lesion areas identified by the proposed CNNRF and the five clinical indicators. An excellent agreement was observed between the model results and expert opinion.

Popular image annotation tools (e.g., Labelme44 and VOTT45) are used to annotate various images and support common formats, such as Joint Photographic Experts Group (jpg), Portable Network Graphics (png), and Tag Image File Format (tiff); these formats are not used in the DICOM data. Therefore, we developed an annotation platform that does not require much storage space or transformations and can be deployed on a private cloud for security and local sharing. Our eyes are not highly sensitive to grayscale images in regions with high average brightness46, resulting in relatively low identification accuracy. The proposed pseudo-color method increased the information content of the medical images and facilitated the identification of the details. PCA has been widely used for feature extraction and dimensionality reduction in image processing47. We used PCA to determine the feature space of the sub-

datasets. Each image in a specified sub-dataset was represented as a linear combination of the eigenvectors. Since the eigenvectors describe the most informative regions in the medical images, they are considered a representation of each sub-dataset. We visualized the top-five eigenvectors of each sub-dataset using an intuitive method.

The CNNCF is a modular framework consisting of two stages that were trained with different optimization goals and controlled by the control gate block. Each stage consisted of multiple residual blocks (ResBlock-A and ResBlock-B) that retained the features in the different layers, thereby preventing degradation of the model. The design of the control gate block was inspired by the synaptic frontend structure in the nervous system. We calculated the score of the optimization target, and a score above a predefined threshold was acceptable. If the times of the neurotransmitter were above another predefined threshold, the control gate was opened to let the features information pass. The framework was trained in a step-by-step manner. Training occurred at each stage for a specified goal, and the second stage used the features extracted by the first stage, thereby reusing the features and increasing the convergence speed of the second stage. The CNNCF exhibited excellent performance for identifying the COVID–19 cases automatically in the X-data and CT-data. Different from traditional machine learning methods, the CNNCF was trained in an end-to-end manner, which ensured the flexibility of the framework for different datasets without much adjustment.

We adopted a knowledge distillation method in the training phrase; a small model (called a student network) was trained to mimic the ensemble of multiple models (called teacher networks) to obtain a small model with high performance. In the distillation process, knowledge was transferred from the teacher networks to the student network to minimize knowledge loss. The target was the output of the teacher networks; these outputs were called soft labels. The student network also learned from the ground-truth labels (also called hard labels), thereby minimizing the knowledge loss from the student networks, whose targets were the hard labels. Therefore, the overall loss function of the student network incorporated both knowledge distillation and knowledge loss from the student networks. After the student network had been well-trained, the task of the teacher networks was complete, and the student model could be used on a regular computer with a fast speed, which is suitable for hospitals without extensive computing resources. As a result of the knowledge distillation method, the CNNCF achieved high performance with a few parameters in the teacher network.

The CNNRF is a modular framework consisting of one stage II sub-framework and one regressor block to handle the regression task. In the regressor block, we used skip connections that consisted of a convolution layer with multiple 1 x 1 convolution kernels for retaining the features extracted by the stage II sub-framework while improving the nonlinear representation ability of the regressor block. We made use of flexible blocks to achieve good performance for the classification and regression tasks, unlike traditional machine learning methods, which are commonly used for either of these tasks.

Five statistical indexes, including sensitivity, specificity, precision, kappa coefficient, and F1 were used to evaluate the performance of the CNNCF. The sensitivity is related to the positive detection rate and is of

great significance in the diagnostic testing of COVID–19. The specificity refers to the ability of the model to correctly identify patients with the disease. The precision indicates the ability of the model to provide positive prediction. The kappa demonstrates the stability of the model's prediction. The F1 is the harmonic mean of the precision and sensitivity. Good performance was achieved by the CNNCF based on the five statistical indices for the multi-modal image datasets (X-data and CT-data). The consistency between the model results and the expert evaluation was determined using McNemar's test. The good performance demonstrated the model's capacity of learning from the experts using the labels of the image data and mimicking the experts in diagnostic decision-making. The ROC and PRC of the CNNCF were used to evaluate the performance of the classification model48. The ROC is a probability curve that shows the trade-off between the true positive rate (TPR) and false positive rate (FPR) using different thresholds settings. The AUROC provides a measure of separability and demonstrated the discriminative capacity of the classification model. The larger the AUROC, the better the performance of the model is for predicting the true positive (TP) and true negative (TN) cases. The PRC shows the trade-off between the TPR and the positive predictive value (PPV) using different thresholds settings. The larger the AUPRC, the higher the capacity of the model is to predict the TP cases. In our experiments, the CNNCF achieved high scores for both the AUPRC and AUROC (>99%) for the X-data and CT-data.

Five statistical indexes, including sensitivity, specificity, precision, kappa coefficient, and F1 were used to evaluate the performance of the CNNCF. The sensitivity is related to the positive detection rate and is of great significance in the diagnostic testing of COVID–19. The specificity refers to the ability of the model to correctly identify patients with the disease. The precision indicates the ability of the model to provide positive prediction. The kappa demonstrates the stability of the model's prediction. The F1 is the harmonic mean of the precision and sensitivity. Good performance was achieved by the CNNCF based on the five statistical indices for the multi-modal image datasets (X-data and CT-data). The consistency between the model results and the expert evaluation was determined using McNemar's test. The good performance demonstrated the model's capacity of learning from the experts using the labels of the image data and mimicking the experts in diagnostic decision-making. The ROC and PRC of the CNNCF were used to evaluate the performance of the classification model. The ROC is a probability curve that shows the trade-off between the true positive rate (TPR) and false positive rate (FPR) using different thresholds settings. The AUROC provides a measure of separability and demonstrated the discriminative capacity of the classification model. The larger the AUROC, the better the performance of the model is for predicting the true positive (TP) and true negative (TN) cases. The PRC shows the trade-off between the TPR and the positive predictive value (PPV) using different thresholds settings. The larger the AUPRC, the higher the capacity of the model is to predict the TP cases. In our experiments, the CNNCF achieved high scores for both the AUPRC and AUROC (>99%) for the X-data and CT-data.

DL has made significant progress in numerous areas in recent years and has provided best-performance solutions for many tasks. In areas that require high interpretability, such as autonomous driving and medical diagnosis, DL has disadvantages because it is a black-box approach and lacks good interpretability. The strong correlation obtained between the CNNCF output and the experts' evaluation suggested that the mechanism of the proposed CNNCF is similar to that used by humans analyzing

images. The combination of the visual interpretation and the correlation analysis enhanced the ability of the framework to interpret the results, making it highly reliable. The CNNCF has a promising potential for clinical diagnosis considering its high performance and hybrid interpretation ability. We have explored the potential use of the CNNCF for clinical diagnosis with the support of the Beijing Youan hospital (which is an authoritative hospital for the study of infectious diseases and one of the designated hospitals for COVID–19 treatment) using both real data after privacy masking and input from experts under experimental conditions and provided a suitable schedule for assisting experts with the radiography analysis. However, medical diagnosis in a real situation is more complex than in an experiment. Therefore, further studies will be conducted in different hospitals with different complexities and uncertainties to obtain more experience in multiple clinical use cases with the proposed framework.

The objective of this study was to use statistical methods to analyze the relationship between salient features in images and expert evaluations and test the discriminative ability of the model. The CNNRF can be considered a cross-modal prediction model, which is a challenging research area that requires more attention because it is closely related to associative thinking and creativity. In addition, the correlation analysis might be a possible optimization direction to improve the interpretability performance of the classification model using DL.

In conclusion, we proposed a complete framework for computer-aided diagnosis of COVID–19, including data annotation, data preprocessing, model design, correlation analysis, and assessment of the model's interpretability. We developed a pseudo-color tool to convert the grayscale medical images to color images to facilitate image interpretation by the experts. We developed a platform for the annotation of medical images characterized by high security, local sharing, and expandability. We designed a simple data preprocessing method for converting multiple types of images (X-data, CT-data) to three-channel color images. We established a modular CNN-based classification framework with high flexibility and wide use cases and consisting of the ResBlock-A, ResBlock-B, and Control Gate Block. A knowledge distillation method was used as a training strategy for the proposed classification framework to ensure high performance with fast inference speed. A CNN-based regression framework that required minimal changes to the architecture of the classification framework was employed to determine the correlation between the lesion area images of patients with COVID–19 and the five clinical indicators. The three evaluation indices (F1, kappa, specificity) of the classification framework were similar to those of the respiratory resident and the emergency resident and slightly higher than that of the respiratory intern. We visualized the salient features that contributed most to the CNNCF output in a heatmap for easy interpretability of the CNNCF. The proposed CNNCF computer-aided diagnosis method demonstrated a promising potential and is highly suitable for the automatic diagnosis of COVID–19 in clinical practice. There is also a potential for broader applicability of the proposed method. Once the method has been improved, it might be used in other diagnostic decision-making scenarios (lung cancer, liver cancer, etc.) using medical images. The expertise of a specialist will be required in clinical cases in future scenarios. However, we are optimistic about the potential of using DL methods in intelligent medicine and expect that many people will benefit from the advanced technology.

Methods

Data collection. Multi-modal image data were used in this study, including X-data and CT-data. The X-data of COVID–19 and the normal cases were obtained from two public datasets (COVID19 chest X-ray dataset, Kaggle RSNA). The X-data of the COVID–19 cases included 212 cases in different time periods (from Feb. 16 to Apr. 19, 2020) and from different countries (Vietnam, China, Canada, USA, UK, Korea, Italy, Australia, Egypt, Spain, German, and Belgium). The X-data of the normal cases (5000 cases) were randomly chosen from the RSNA dataset, which can be downloaded from this website36. The CT-data of the COVID–19 (95 cases) and influenza cases (50 cases) were front-line clinical data provided by the Beijing Youan hospital, which is an authoritative hospital for the study of infectious diseases and one of the designated hospitals for COVID–19 treatment. The CT-data of the normal cases (120 cases) were obtained from the LUNA–16 dataset. All data followed current laws and regulations and had been processed using data-masking methods to ensure the privacy of the individuals. The processed data did not contain personal information (e.g., name, ID number, phone number), and the files were randomly renamed using pseudo-naming methods. A self-developed annotation tool was used to annotate the lesion areas in the images of the COVID–19 and influenza cases, and the annotation was confirmed by two experts.

Dataset splitting. We split the X-data, CT-data, and clinical indicator data into train-val datasets and test datasets. The details are shown in Table 1.

1. **X-data**: A total of 5212 X-data from 5212 cases (5000 normal cases and 212 COVID–19 cases) were named XDS. We used the train-test-split function (TTSF) of the scikit-learn library to split the XDS into a train-val dataset (XTS) and a test dataset (XVS) as follows: 4850 normal cases of the 5000 normal cases were randomly selected from the XDS to comprise part of the XTS; 150 COVID–19 cases of the 212 COVID–19 cases were randomly selected from the XDS to comprise another part of the XTS. The remainder of the XDS constituted the test dataset XVS (62 COVID–19 cases and 150 normal cases). Each case was either part of the XTS or the XVS.
2. **CT-data**: A total of 100,521 of the 265 cases of the CT-data (120 normal cases, 50 influenza cases, and 95 COVID–19 cases) were obtained from the hospital and the LUNA–16 dataset; this represented the CTDS. We used the TTSF to split the CTDS into a train-val dataset (CTTS) and a test dataset (CTVS) as follows: 100 normal cases of the 120 normal cases, 35 influenza cases of the 50 influenza cases, and 75 COVID–19 cases of the 90 COVID–19 cases were randomly selected from the CTDS and constituted the CTTS. The rest of the CTDS data represented the CTVS (20 normal cases, 15 influenza cases, and 20 COVID–19 cases). Each case was either part of the CTTS or the CTVS.
3. **Clinical indicator data**: A total of 95 data pairs from the 95 COVID–19 cases (369 images of the lesion area and 95 5 clinical indicators) were collected from the hospital; this represented the CADS. The CADS was randomly split into a train-val dataset (CATS) (with 75 data pairs from the 75 COVID–19 cases) and a test dataset (CAVS) (with 20 data pairs from 20 COVID–19 cases) using the TTSF.

Image preprocessing. All image data (X-data and CT-data) in the DICOM format were loaded using the Pydicom library (version 1.4.0) and processed as arrays using the Numpy library (version 1.16.0).

1. **X-data**: The two-dimensional array (x-axis and y-axis) of the image of the X-data (size of 512 × 512) was normalized to pixel values of 0–255 and stored in png format using the OpenCV library. Each preprocessed image was resized to 512 × 512 and had 3 channels.
2. **CT-data**: The array of the CT-data was three-dimensional (x-axis, y-axis, and z-axis), and the length of the z-axis was approximately 300, which represented the number of image slices. Each image slice was two-dimensional (x-axis and y-axis, size of 512 x 512). As shown in Supplementary Fig. 2, the array of the image was divided into three groups in the z-axis direction, and each group contained 100 image slices (each case was resampled to 300 image slices). The image slices in each group were processed using a window center of −600 and a window width of 2000 to extract the lung tissue. The images of the CT-data with >300 image slices were normalized to pixel values of 0–255 and stored in npy format using the Numpy library. A convolution filter was applied with three 1 x 1 convolution kernels to preprocess the CT-data; the image size was 512 x 512, with 3 channels.

Annotation tool for medical images. The server program of the annotation tool was deployed in a computer with large network bandwidth and abundant storage space. The client program of the annotation tool was deployed in the office computer of the experts, who were given unique user IDs for login. The interface of the client program had a built-in image viewer with a window size of 512 x 512 and an export tool for obtaining the annotations in text format. Multiple drawing tools were provided to annotate the lesion area in the images, including a rectangle tool for drawing a bounding box around the target, a polygon tool for drawing the outline of the target, and a circle tool for drawing a curved outline of the target. Multiple categories could be defined and assigned to the target areas. All annotations were stored in a structured query language (SQL) database, and the export tool was used to export the annotations to two common file formats (comma-separated values (csv) and JavaScript object notation (json)). The experts could share the annotation results. Since the size of the X-data and the slice of the CT-data were identical, the annotations for both data were performed in with the annotation tool. Here we use one image slice of the CT-data as an example to demonstrate the annotation process. In this study, two experts were asked to annotate the medical images. The normal cases were reviewed and confirmed by the experts. The abnormal cases, including the COVID–19 and influenza cases, were annotated by the experts. Bounding boxes of the lesion areas in the images were annotated using the annotation tool. In general, each case contained 2 to 5 slices with annotations. The cases with the annotated slices were considered positive cases, and each case was assigned to a category (COVID–19 case or influenza case).

Model architecture and training. In this study, we proposed a modular CNNCF to identify the COVID–19 cases in the medical images and a CNNRF to determine the relationships between the lesion areas in the medical images and the five clinical indicators of COVID–19. Both proposed frameworks consisted of two units (ResBlock-A and ResBlock-B). The CNNCF and CNNRF had unique units,

namely the control gate block and regressor block, respectively. Both frameworks were implemented using two NVIDIA GTX 1080TI graphics cards and the open-source PyTorch framework.

1. **ResBlock-A.** As discussed in Ref.49, the residual block is a CNN-based block that allows the CNN models to reuse features, thus accelerating the training speed of the models. In this study, we developed a residual block (ResBlock-A) that utilized a skip connection for retaining features in different layers in the forward propagation. This block (Fig. 6-a) consisted of a multiple-input multiple-output structure with two branches (an upper branch and a bottom branch), where input 1 and input 2 have the same size, but the values may be different. In contrast, output 1 and output 2 had the same size, but output 1 did not have a ReLu layer. The upper branch consisted of a max-pooling layer (Max-Pooling), a convolution layer (Conv 1 x 1) and a batch norm layer (BN). The Max-Pooling had a kernel size of 3 x 3 and a stride of 2 to downsample the input 1 for retaining the features and ensuring the same size as the output layer before the element-wise add operation was conducted in the bottom branch. The Conv 1 x 1 consisted of multiple 1 x 1 convolution kernels with the same number as that in the second convolution layer in the bottom branch to adjust the number of channels. The BN used a regulation function to ensure the input in each layer of the model followed a normal distribution with a mean of 0 and a variance of 1. The bottom branch consisted of two convolution layers, two BN layers, and two ReLu layers. The first convolution layer in the bottom branch consisted of multiple 3 x 3 convolution kernels with a stride of 2 and a padding of 1 to reduce the size of the feature maps when local features were obtained. The second convolution layer in the bottom branch consisted of multiple 3 x 3 convolution kernels with a stride of 1 and a padding of 1. The ReLu function was used as the activation function to ensure a nonlinear relationship between the different layers. The output of the upper branch and the output of the bottom branch after the second BN were fused using an element-wise add operation. The fused result was output 1, and the fused result after the ReLu layer was output 2.
2. **ResBlock-B.** The ResBlock-B (Fig. 6-b) was a multiple-input single-output block that was similar to the ResBlock-A, except that there was no output 1. The value of the stride and padding in each layer of the ResBlock-A and ResBlock-B could be adjusted using hyper-parameters based on the requirements.
3. **Control Gate Block.** As shown in Fig. 6-c, the Control Gate Block was a multiple-input single-output block consisting of a predictor module, a counter module, and a synapses module to control the optimization direction while controlling the information flow in the framework. The pipeline of the predictor module is shown in Supplementary Fig. 3-a, where the Input S1 is the output of the ResBlock-B. The Input-S1 is then flattened to a one-dimensional feature vector as the input of the linear layer. The output of the linear layer was converted to a probability of each category using the softmax function. A sensitivity calculator used the V_{pred} and V_{true} as inputs to calculate the TP, TN, FP, and false negative (FN) rates to calculate the sensitivity. The sensitivity calculation was followed by a step function to control the output of the predictor. The th_s was a threshold value; if the calculated sensitivity was greater or equal to th_s, the step function output

1; otherwise, the output was 0. The counter module was a conditional counter, as shown in Supplementary Fig. 3-b. If the input *n* was zero, the counter was cleared and set to zero. Otherwise, the counter increased by 1. The output of the counter was *num.* The synapses block mimicked the synaptic structure, and the input variable *num* was similar to a neurotransmitter, as shown in Supplementary Fig. 3-c. The input *num* was the incoming parameter of the step function. The th_s was a threshold value; if the input *num* was greater or equal to th_s, the step function output 1; otherwise, it output 0. An element-wise multiplication was performed between the input S1 and the output of the synapses block. The multiplied result was passed on to a discriminator. If the sum of each element in the result was not zero, the Input S1 was passed on to the next layer. Otherwise, the input S1 information was not passed on.

4. **Regressor block.** The regressor block consisted of multiple linear layers, a convolution layer, a BN layer, and a ReLu layer, as shown in Fig. 6-d. A skip-connection architecture was adopted to retain the features and increase the ability of the block to represent nonlinear relationships. The convolution block in the skip-connection structure was a convolution layer with multiple numbers of 1 x 1 convolution kernels. The number of the convolution kernels was the same as that of the output size of the second linear layer to ensure the consistency of the vector dimension. The input size and output size of each linear layer were adjustable to be applicable to actual cases.

Based on the four blocks, two frameworks were designed for the classification task and regression task, respectively.

1. **Classification framework.** The CNNCF consisted of stage I and stage II, as shown in Fig. 5-a. Stage I was duplicated Q times in the framework (in this study, Q = 1). It consisted of multiple ResBlock-A with a number of M (in this study, M = 2), one ResBlock-B, and one Control Gate Block. Stage II consisted of multiple ResBlock-A with a number of N (in this study, N = 2) and one ResBlock-B. The weighted cross-entropy loss function was used and was minimized using the SGD optimizer with a learning rate of a1 (in this study, a1 = 0.01). A warm-up strategy50 was used in the initialization of the learning rate for a smooth training start, and a reduction factor of b1 (in this study, b1 = 0.1) was used to reduce the learning rate after every c1 (in this study, c1 = 10) training epochs. The model was trained for d1 (in this study, d1 = 40) epochs, and the model parameters saved in the last epoch was used in the test phrase.
2. **Regression framework.** The CNNRF (Fig. 5-b) consisted of two parts (stage II and the regressor). The inputs to the regression framework were the images of the lesion areas, and the output was the corresponding vector with five dimensions, representing the five clinical indicators (all clinical indicators were normalized to a range of 0 to 1). The stage II structure was the same as that in the classification framework, except for some parameters. The loss function was the MSE loss function, which was minimized using the SGD optimizer with a learning rate of a2 (in this study, a2 = 0.01). A warmup strategy was used in the initialization of the learning rate for a smooth training start, and a reduction factor of b2 (in this study, b2 = 0.1) was used to reduce

the learning rate after every c2 (in this study, c2 = 50) training epochs. The framework was trained for d2 (in this study, d2 = 200) epochs, and the model parameters saved in the last epoch were used in test phrase.

Training strategies of the classification framework. The training strategies and hyper-parameters of the classification framework were as follows. We adopted a knowledge distillation method (Fig. 7) to train the CNNCF as a student network with one stage I block and one stage II block, each of which contained two ResBlock-A. Four teacher networks (the hyper-parameters are provided in the Supplementary Table 5) with the proposed blocks were trained on the XTS and the CTTS using a 5-fold cross-validation method. All networks were initialized using the Xavier initialization method. The initial learning rate was 0.01, and the optimization function was the SGD. The CNNCF was trained using the image data and the label, as well as the fused output of the teacher networks. Supplementary Fig. 5 shows the details of the knowledge distillation method.

Gradient-weighted class activation maps. Grad-CAM51 in the Pytorch framework was used to visualize the salient features that contributed the most to the prediction output of the model Given a target category, the Grad-CAM performed back-propagation to obtain the final CNN feature maps and the gradient of the feature maps; only pixels with positive contributions to the specified category were retained through the ReLU function. The Grad-CAM method was used for all test dataset (XVS and CTVS) in the CNNCF without changing the framework structure to obtain a visual output of the framework's high discriminatory ability.

Statistical indices and empirical distribution of the test results. We used multiple statistical indices and empirical distributions to assess the performance of the proposed frameworks. The equations of the statistical indices are shown in Supplementary Fig. 6.

1. **Statistical indices to evaluate the classification framework.** Multiple evaluation indicators (PRC, ROC, AUPRC, AUROC, sensitivity, specificity, precision, kappa index, and F1 with a fixed threshold) were computed for a comprehensive and accurate assessment of the classification framework. Multiple threshold values were in the range from 0 to 1 with a step value of 0.005 to obtain the ROC and PRC curves. The PRC showed the relationship between the precision and the sensitivity (or recall), and the ROC indicated the relationship between the sensitivity and specificity. The two curves reflected the comprehensive performance of the classification framework. The kappa index is a statistical method for assessing the degree of agreement between different methods. In our use case, the indicator was used to measure the stability of the method. The F1 score is a harmonic average of precision and sensitivity and considers the FP and FN. The bootstrapping method was used to calculate the empirical distribution of each indicator. The detailed calculation process was as follows: we conducted random sampling with replacement to generate 1000 new test datasets with the same number of samples as the original test dataset. The evaluation indicators were calculated to determine the distributions. The results were displayed in boxplots (Fig. 4 and Supplementary Fig. 4).

2. **Statistical indices to evaluate the regression framework.** Multiple evaluation indicators (MSE, RMSE, MAE, R2, and PCC) were computed for a comprehensive and accurate assessment of the regression framework. The MSE was used to calculate the deviation between the predicted and true values. The RMSE was the square root of the MSE result. The two indicators show the accuracy of the model prediction. The R2 was used to assess the goodness-of-fit of the regression framework. The r was used to assess the correlation between two variables built from the regression framework. The indicators were calculated using the open source tools scikit-learn and the scipy library.

References

1. Sanche, S. *et al.* High contagiousness and rapid spread of severe acute respiratory syndrome coronavirus 2. *Emerging infectious diseases* **26** (2020).
2. Mahase, E. Covid-19: Who declares pandemic because of “alarming levels” of spread, severity, and inaction. *BMJ* **368** (2020).
3. Organization, H. Who covid-19 dashboard. https://covid19.who.int/. Accessed May 5, 2020.
4. Goodman, D. World economy faces $5 trillion hit. https://www.bloomberg.com/ news/articles/2020-04-08/world-economy-faces-5-trillion-hit- that-is-like-losing-japan. Accessed May 5, 2020
5. for Disease Control, C. & Prevention. Clinical specimens: Novel coronavirus (2019- ncov). https://www.cdc.gov/coronavirus/2019-ncov/lab/guidelines- clinical-specimens. Accessed May 5, 2020
6. Organization, H. *et al.* Laboratory testing for coronavirus disease 2019 (covid-19) in suspected human cases: interim guidance, 2 march 2020. Tech. Rep., World Health Organization (2020).
7. Organization, H. *et al.* Laboratory biosafety guidance related to coronavirus disease 2019 (covid-19): interim guidance, 12 february 2020. Tech. Rep., World Health Organization (2020).
8. Yang, *et al.* Laboratory diagnosis and monitoring the viral shedding of 2019-ncov infections. *MedRxiv* (2020).
9. Fang, *et al.* Sensitivity of chest ct for covid-19: comparison to rt-pcr. *Radiology* 200432 (2020).
10. Wong, Y. F. *et al.* Frequency and distribution of chest radiographic findings in covid-19 positive patients. *Radiology* 201160 (2020).
11. McIntosh, Coronavirus disease 2019 (covid-19): Epidemiology, virology, clinical features, diagnosis, and prevention. uptodate 2020 (2020).
12. Organization, H. *et al.* Clinical management of severe acute respiratory infection when novel coronavirus (2019-ncov) infection is suspected: interim guidance. In *Clinical management of severe acute respiratory infection when novel coronavirus (2019-nCoV) infection is suspected: Interim guidance*, 21–21 (2020).
13. Organization, H., Organization, W. H. *et al.* Report of the who-china joint mission on coronavirus disease 2019 (covid-19) (2020).

14. Razzak, M. I., Naz, S. & Zaib, A. Deep learning for medical image processing: Overview, challenges and the future. In *Classification in BioApps*, 323–350 (Springer, 2018).
15. Attwood, D. & Sakdinawat, A. *X-rays and extreme ultraviolet radiation: principles and applications* (Cambridge university press, 2017).
16. Goldman, L. Principles of ct and ct technology. *Journal of nuclear medicine technology* **35**, 115–128 (2007).
17. Dougherty, G. *Digital image processing for medical applications* (Cambridge University Press, 2009).
18. Smith-Bindman, R., Miglioretti, D. L. & Larson, E. B. Rising use of diagnostic medical imaging in a large integrated health system. *Health affairs* **27**, 1491–1502 (2008).
19. Udugama, B. *et al.* Diagnosing covid-19: The disease and tools for detection *ACS nano* (2020)
20. Kim, H., Hong, H. & Yoon, H. Diagnostic performance of ct and reverse transcriptase-polymerase chain reaction for coronavirus disease 2019: a meta-analysis. *Radiology* 201343 (2020).
21. Sermanet, , Chintala, S. & LeCun, Y. Convolutional neural networks applied to house numbers digit classification. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, 3288–3291 (IEEE, 2012).
22. Rawat, & Wang, Z. Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation* **29**, 2352–2449 (2017).
23. Sun, , Xue, B., Zhang, M. & Yen, G. G. Evolving deep convolutional neural networks for image classification. *IEEE Transactions on Evolutionary Computation* (2019).
24. Zhao, -Q., Zheng, P., Xu, S.-t. & Wu, X. Object detection with deep learning: A review.*IEEE transactions on neural networks and learning systems* **30**, 3212–3232 (2019).
25. Silver, D. *et al.* Mastering the game of go without human knowledge. *Nature* **550**, 354–359 (2017).
26. Young, , Hazarika, D., Poria, S. & Cambria, E. Recent trends in deep learning based natural language processing. *ieee Computational intelligenCe magazine* **13**, 55–75 (2018).
27. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778 (2016).
28. Kuruvilla, J. & Gunavathi, K. Lung cancer classification using neural networks for ct *Computer methods and programs in biomedicine* **113**, 202–209 (2014).
29. Rajpurkar, *et al.* Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225* (2017).
30. Gulshan, *et al.* Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *Jama* **316**, 2402–2410 (2016).
31. Baydin, A. G., Pearlmutter, B. A., Radul, A. A. & Siskind, J. M. Automatic differentiation in machine learning: a *The Journal of Machine Learning Research* **18**, 5595–5637 (2017).
32. Lee, G. & Fujita, H. Deep learning in medical image analysis: challenges and applications (2020).
33. Mildenberger, , Eichelberg, M. & Martin, E. Introduction to the dicom standard. *European radiology* **12**, 920–927 (2002).

34. LeCun, , Bengio, Y. & Hinton, G. Deep learning. *nature* **521**, 436–444 (2015).
35. Cohen, J. , Morrison, P. & Dao, L. Covid-19 image data collection. *arXiv 2003.11597*(2020).
36. Rsna pneumonia detection challenge. https://www.kaggle.com/c/rsna- pneumonia-detection-challenge/data. Accessed May 5, 2020.
37. Setio, A. A. *et al.* Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the luna16 challenge. *Medical image analysis* **42**, 1–13 (2017).
38. Johnson, R. An introduction to the bootstrap. *Teaching Statistics* **23**, 49–54 (2001).
39. McNemar, Q. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika* **12**, 153–157 (1947).
40. Fisher, R. A. Theory of statistical estimation. In *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 22, 700–725 (Cambridge University Press, 1925).
41. Soper, H., Young, , Cave, B., Lee, A. & Pearson, K. On the distribution of the correlation coefficient in small samples. appendix ii to the papers of" student" and ra fisher. *Biometrika* **11**, 328–413 (1917).
42. Robbins, H. & Monro, S. A stochastic approximation method. *The annals of mathematical statistics* 400–407 (1951).
43. Glorot, X. & Bengio, Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 249–256 (2010).
44. Wada, labelme: Image polygonal annotation with python. https://github.com/ wkentaro/labelme (2018). Accessed May 5, 2020.
45. group CSE at Microsoft. Vott: Visual object tagging tool. https://github.com/ microsoft/VoTT (2019). Accessed May 5, 2020.
46. Pomerantz, J. R. Perception: *Encyclopedia of cognitive science* (2006).
47. Mudrova, & Procha´zka, A. Principal component analysis in image processing. In *Proceedings of the MATLAB Technical Computing Conference, Prague* (2005).
48. Davis, J. & Goadrich, M. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, 233–240 (2006).
49. He, , Zhang, X., Ren, S. & Sun, J. Identity mappings in deep residual networks. In *European conference on computer vision*, 630–645 (Springer, 2016).
50. Goyal, *et al.* Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677* (2017).
51. Selvaraju, R. R. *et al.* Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, 618– 626 (2017).

Declarations

Acknowledgements The authors would like to thank the Ministry of Science and Technology of the People's Republic of China (Grant No. 2017YFB1400100) and the National Natural Science Foundation of China (Grant No. 61876059) for their support.

Competing Interests The authors declare that they have no competing financial interests.

Correspondence Correspondence and requests for materials should be addressed to Yu Gu. (email: guyu@mail.buct.edu.cn).

Tables

Table 1: Number of cases in the different datasets (X-data, CT-data, clinical indicator data)

Study	X-ray		CT		Clinical	
	Train+Val	Test	Train+Val	Test	Train+Val	Test
Normal	5000	100	100	20	-	-
COVID-19	150	62	75	20	75	20
Influenza	-	-	35	15	-	-
Total	5150	162	210	55	75	20

Table 2: Performance indices of the classification framework (CNNCF) for the XVS data and the average performance of the 7th year respiratory resident (Respira.), the 3rd year emergency resident (Emerg.), and the1st year respiratory intern (Intern).

Study	X-ray		CT		Clinical	
	Train+Val	Test	Train+Val	Test	Train+Val	Test
Normal	5000	100	100	20	-	-
COVID-19	150	62	75	20	75	20
Influenza	-	-	35	15	-	-
Total	5150	162	210	55	75	20

Table 3: Performance indices of the classification framework (CNNCF) for the CTVS data and the average performance of the 7th year respiratory resident (Respira.), the 3rd year emergency resident (Emerg.), and the 1st year respiratory intern (Intern).

	CT(Normal and COVID-19 cases)				CT(Influenza and COVID-19 cases)			
	CNNCF	Respire.	Emerg.	Intern.	CNNCF	Respire.	Emerg.	Intern.
F1	1.0000	1.0000	1.0000	0.9500	1.0000	1.0000	0.8966	0.8000
Kappa	1.0000	1.0000	1.0000	0.9500	1.0000	1.0000	0.8236	0.6500
Specificity	1.0000	1.0000	1.0000	0.9500	1.0000	1.0000	0.9048	0.8500
Sensitivity	1.0000	1.0000	1.0000	0.9500	1.0000	1.0000	0.9286	0.8000
Precision	1.0000	1.0000	1.0000	0.9500	1.0000	1.0000	0.8667	0.8000

Figures

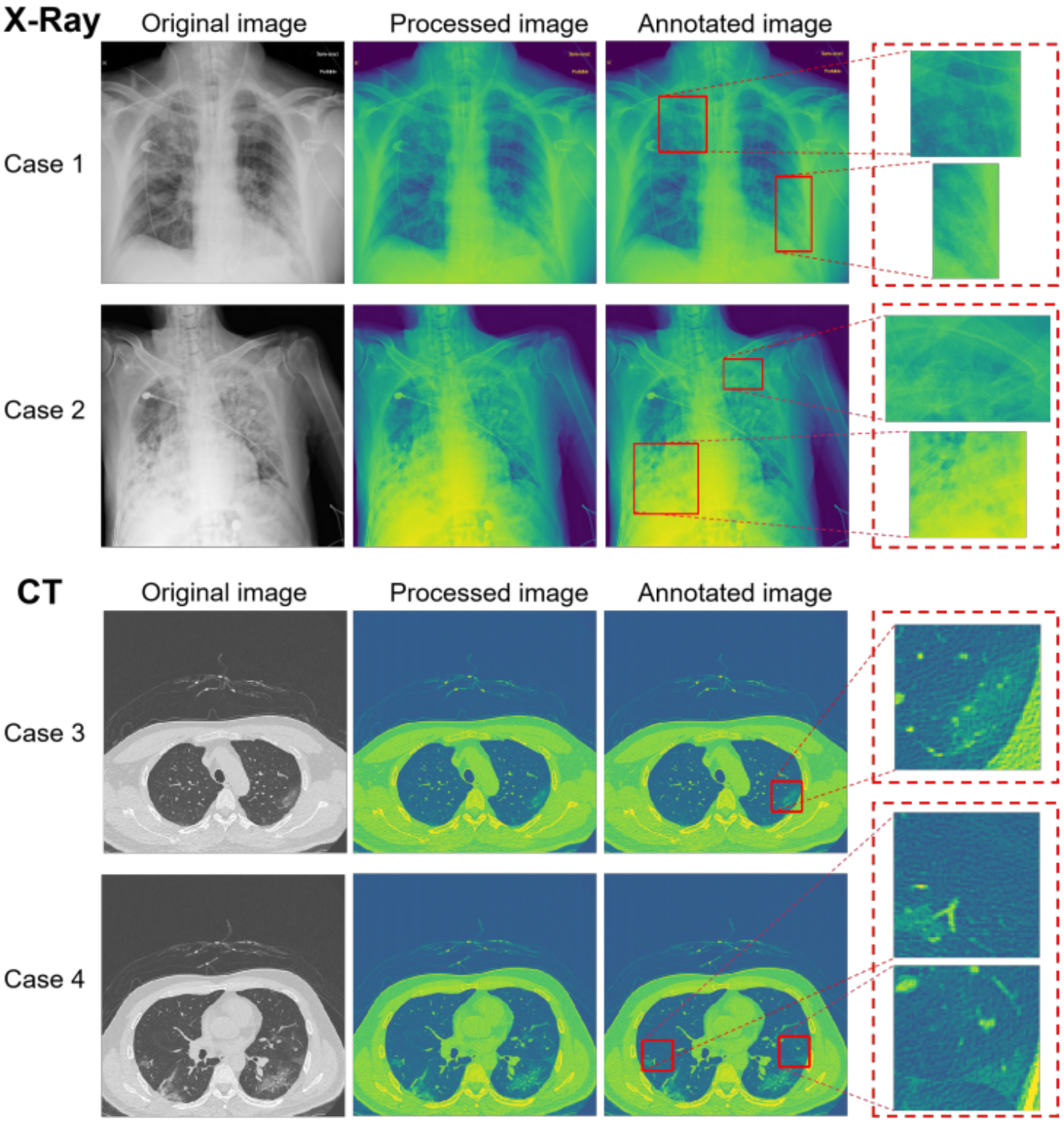

Figure 1

Abnormal examples in the X-ray and CT images. The original grayscale images were transformed into color images using the pseudo-coloring method and were annotated by the experts.

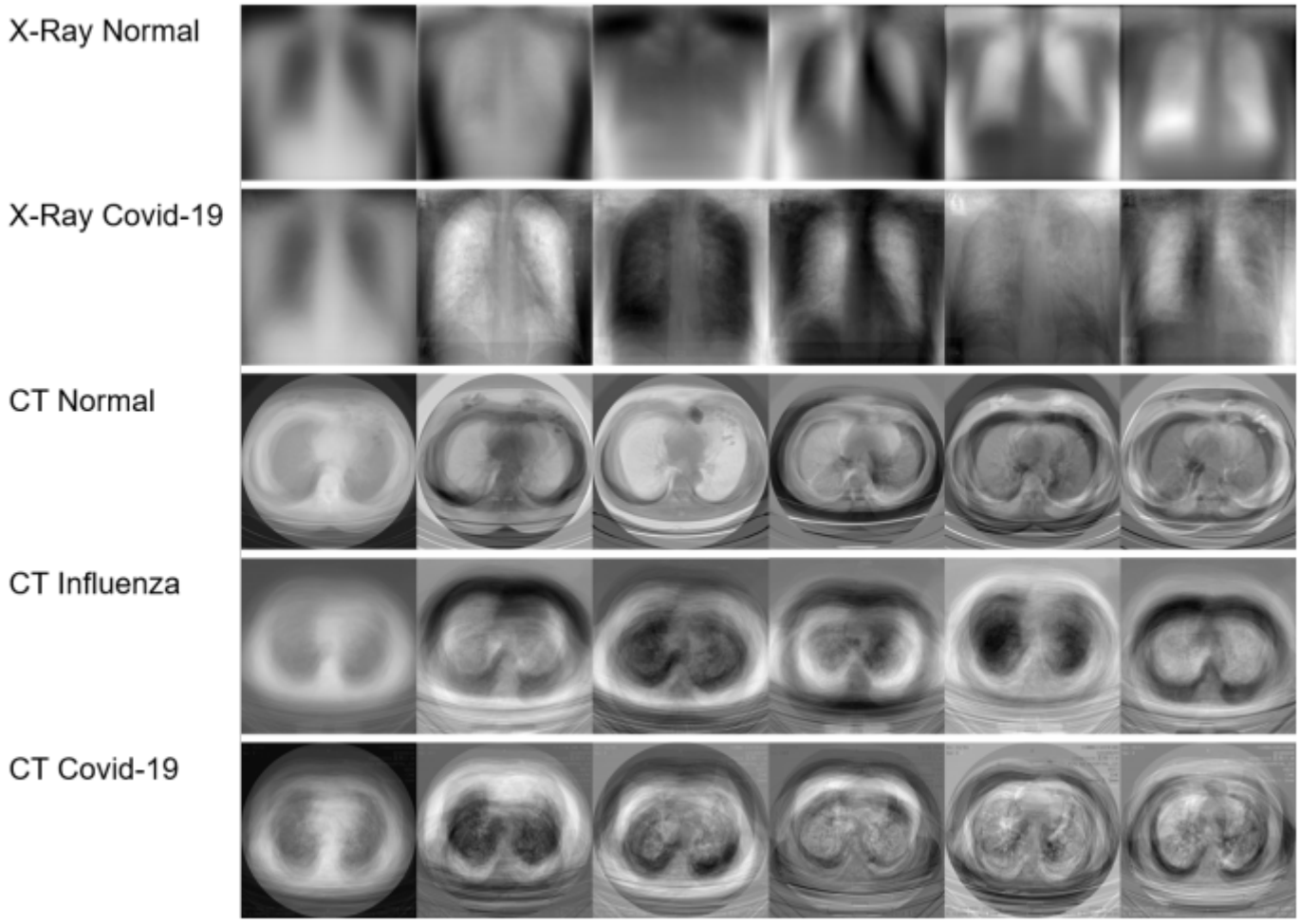

Figure 2

PCA visualization. Mean image and eigenvectors of five different sub-datasets. The first column shows the mean image and the other columns show the eigenvectors. The first row shows the mean image and five eigenvectors of the XNDS; second row: XCDS, third row: CTCS, fourth row: CTIS, last row: CTCS.

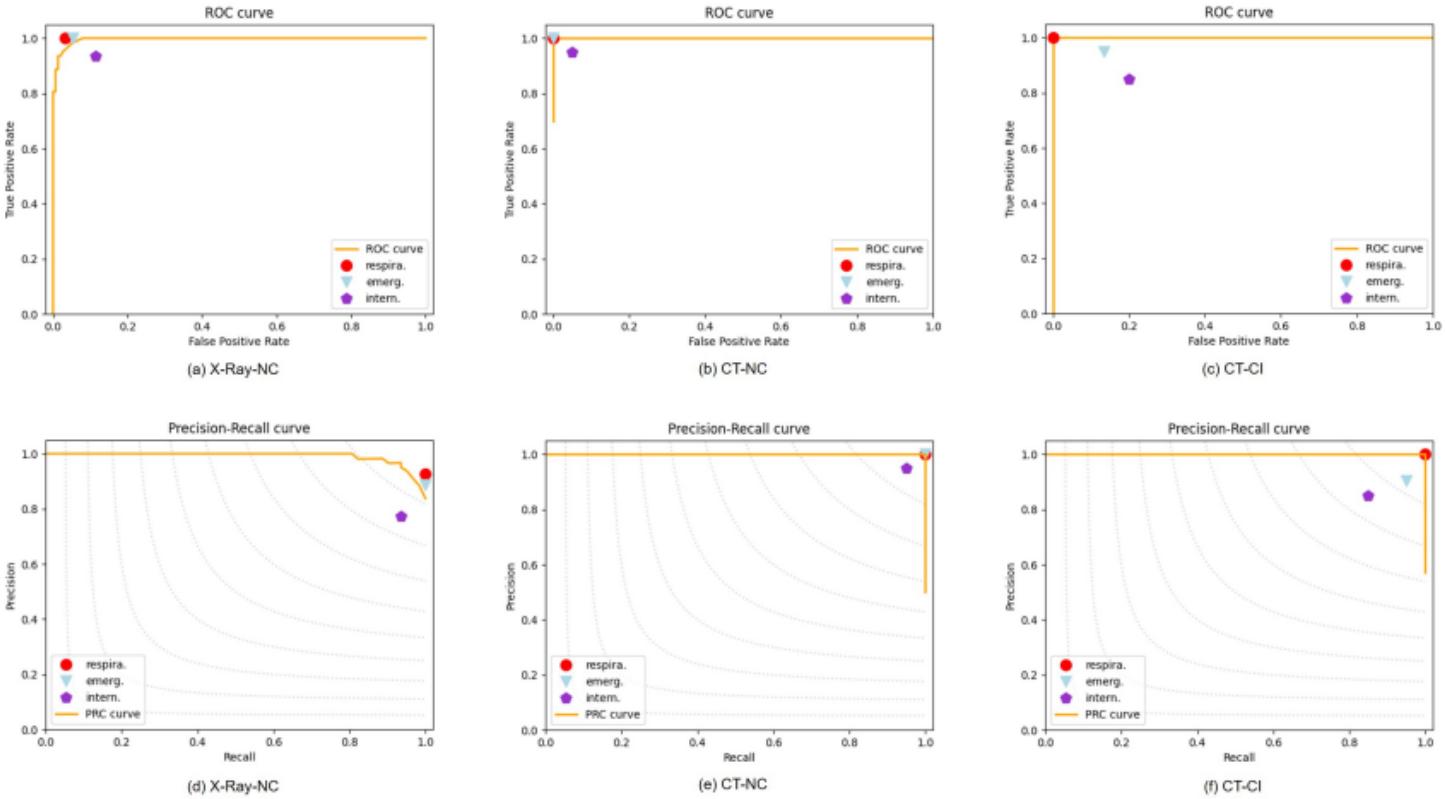

Figure 3

ROC and PRC curves for the CNNCF and the XVS and CTVS. NC indicates that the positive case is a COVID-19 case, and the negative case is normal. CI indicates that the positive case is COVID-19, and the negative case is influenza. The points are the results of experts, cor- responding to the results in Table 2 and Table 3. The background gray dashed curves in the PRC curve correspond to the iso-F1 curves.

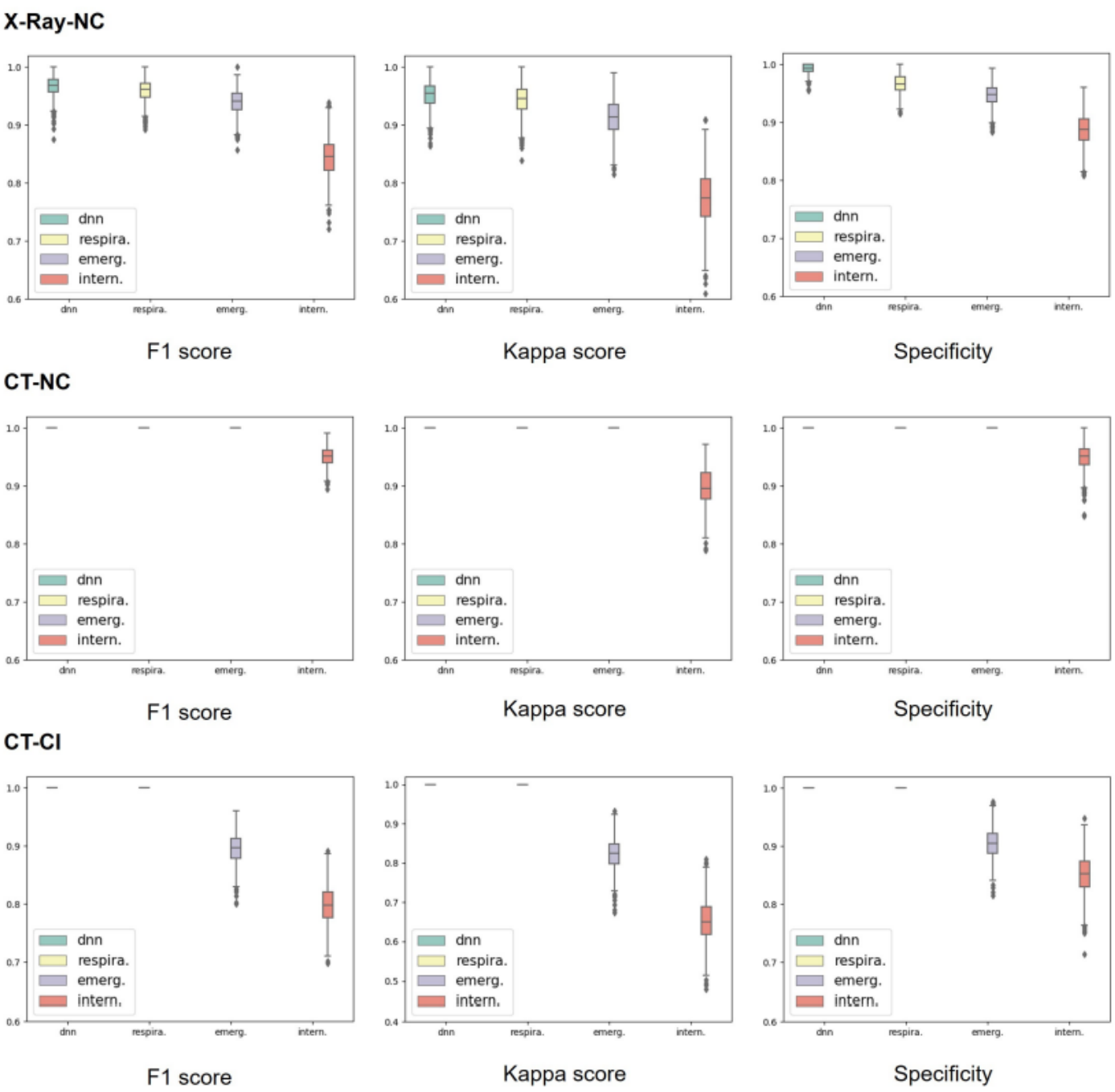

Figure 4

Boxplots of the F1 score, kappa score, and specificity for the CNNCF and expert results for COVID-19 identification. NC indicates that the positive case is a COVID-19 case, and the negative case is normal. CI indicates that the positive case is COVID-19, and the negative case is influenza. Bootstrapping is used to generate 1000 resampled validation sets for the XVS and the CTVS.

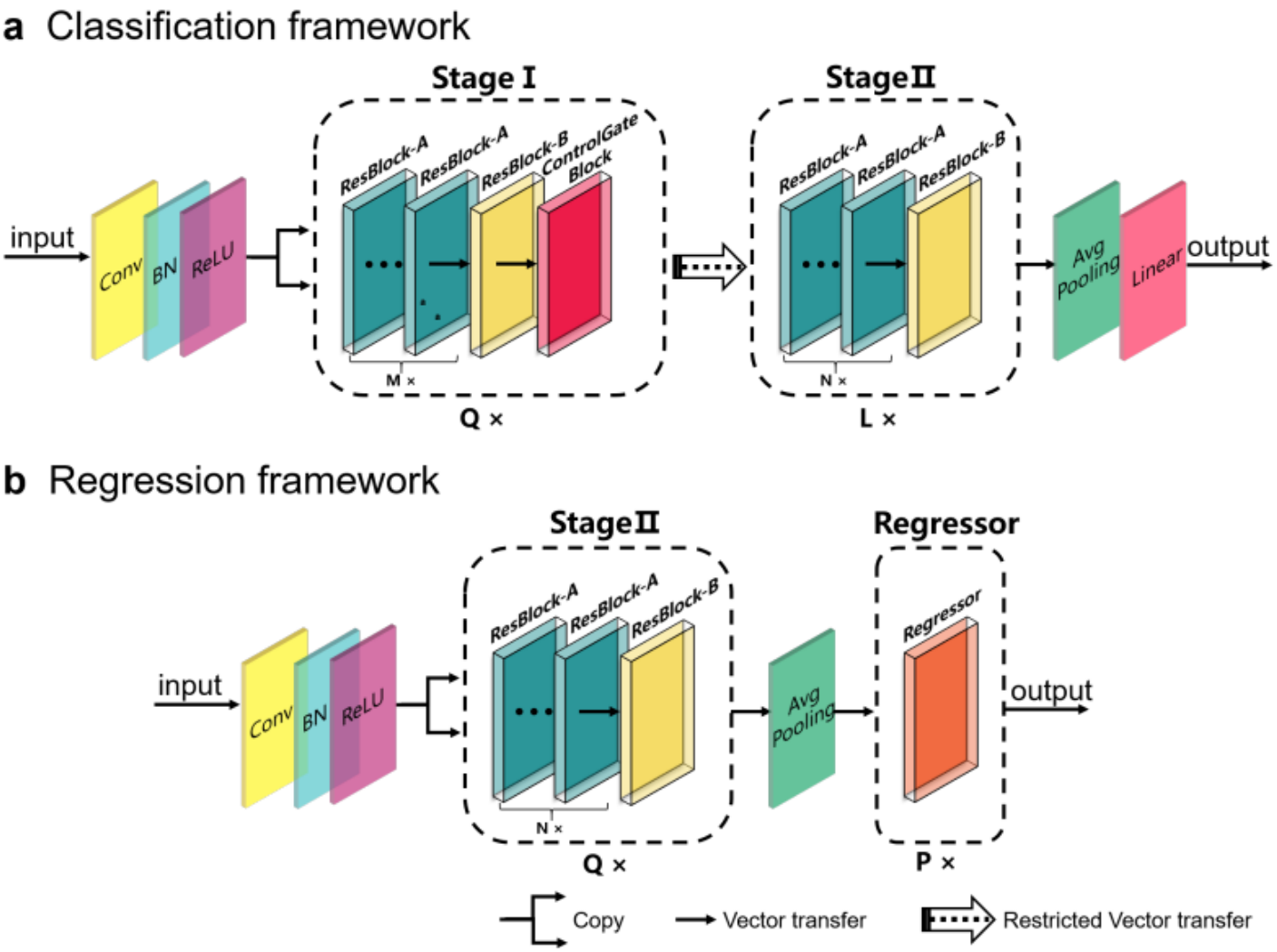

Figure 5

CNN-based frameworks. a is the classification framework for the identification of COVID-19. b is the regression framework for the correlation analysis between the lesion areas and the clinical indicators.

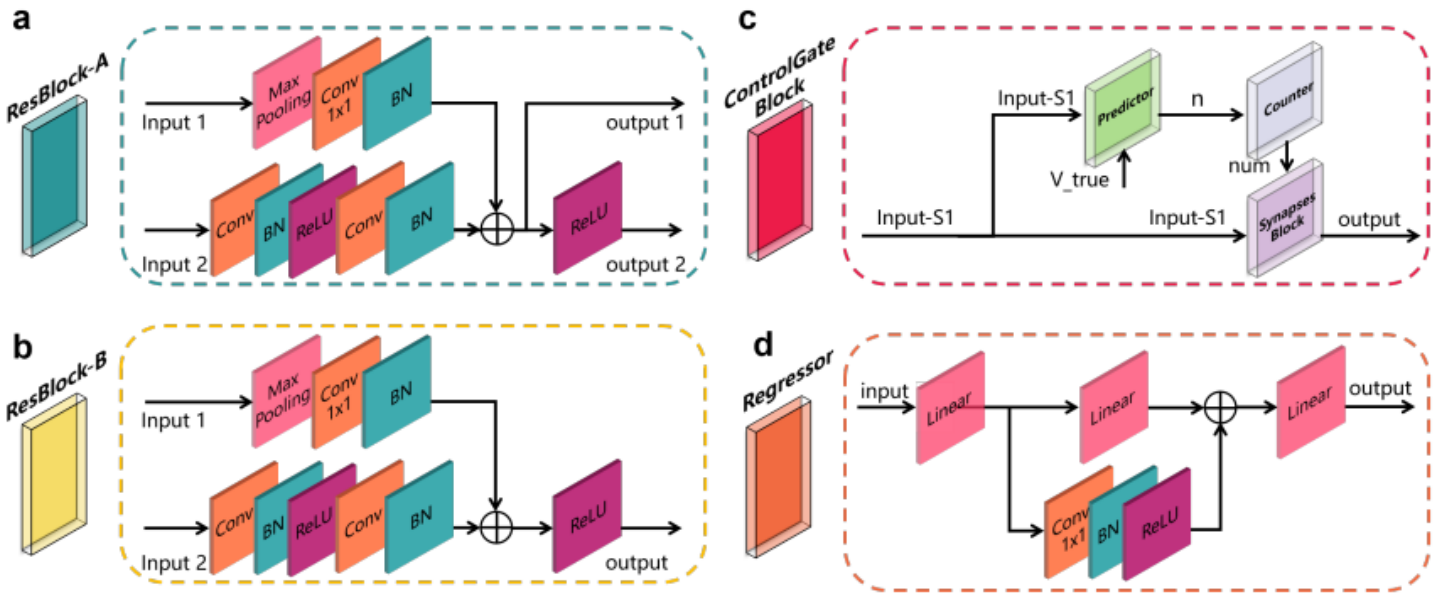

Figure 6

The four units of the proposed framework. a ResBlock-A architecture, containing two convolution layers with 3 x 3 kernels, one convolution layer with a 1 x 1 kernel, three batch normalization layers, two ReLu layers, and one max-pooling layer with a 3 x 3 kernel. b ResBlock-B architecture; the basic unit is the same as the ResBlock-A, except for output 1. c The Control Gate Block has a synaptic-based frontend architecture that controls the direction of the feature map flow and the overall optimization direction of the framework. d The Regressor architecture is a skip-connection architecture containing one convolution layer with 3 x 3 kernels, one batch normalization layer, one ReLu layer, and three linear layers.

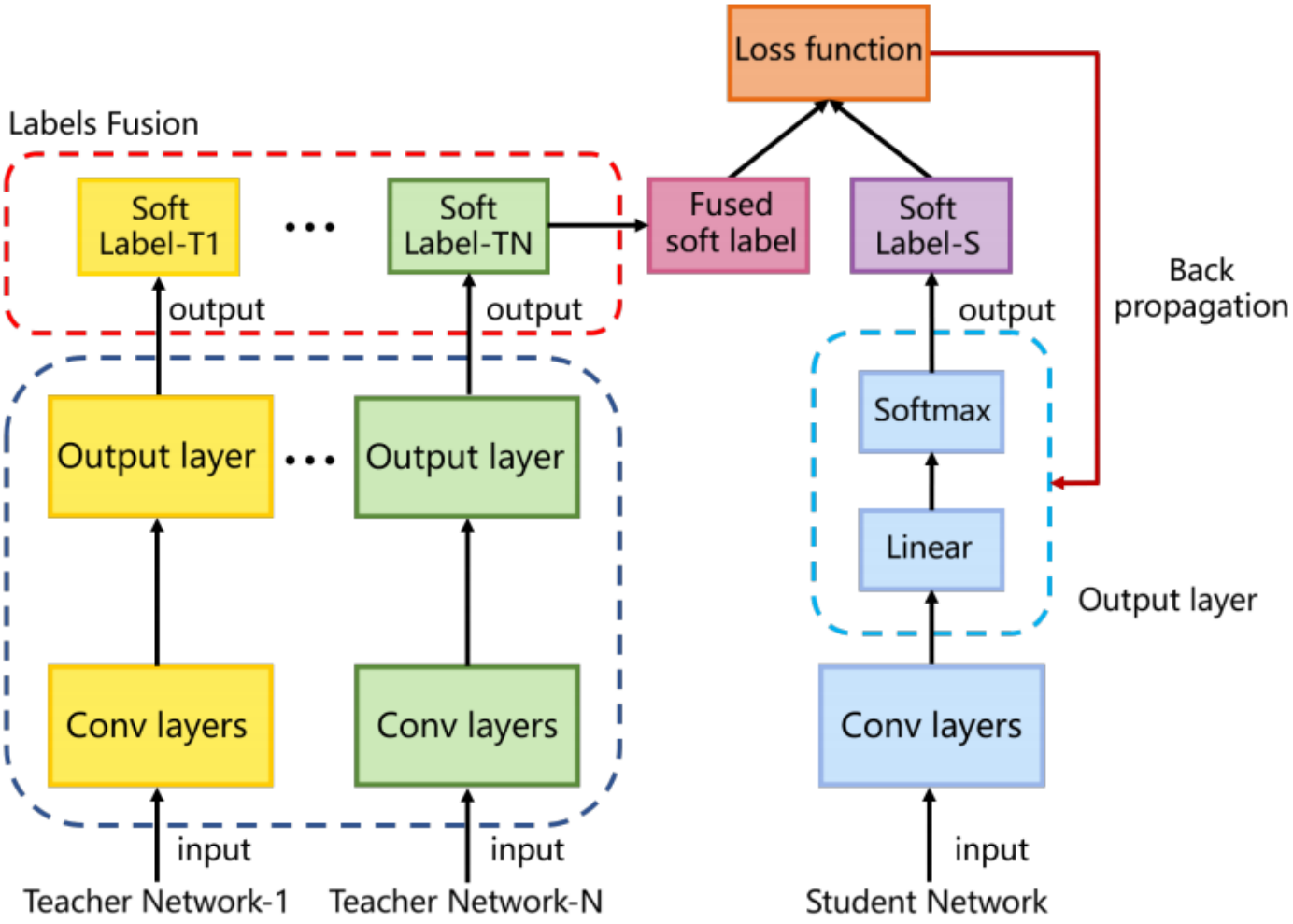

Figure 7

Knowledge distillation consisting of multiple teacher networks and a target student network. The knowledge is transferred from the teacher networks to the student network using a loss function.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- suplementarycomplete.pdf