Accurate and Rapid Diagnosis of COVID-19 Pneumonia with Batch Effect Removal of Chest CT-Scans and Interpretable Artificial Intelligence

Rassa Ghavami¹,², Mehrab Hamidi¹,², Saeed Masoudian¹, Amir Mohseni¹, Hamzeh Lotfalinezhad¹, Mohammad Ali Kazemi³, Behnaz Moradi³, Mahyar Ghafoori⁴, Omid Motamedi⁴, Omid Purnik⁴, Kiara Rezaei-Kalantari⁵, Amirreza Manteghinezhad⁶, Shaghayegh Haghjooy⁶, Fateme Abdoli Nezhad⁷, Ahmad Enhesari⁷, Mohammad Saeed Kheyrkhah⁸, Razieh Eghtesadi⁹, Javid Azadbakht¹⁰, Akbar Aliasgharzadeh⁹, Mohammad Reza Sharif¹¹, Ali Khaleghi¹², Abbas Foroutan¹³, Hossein Ghenaati³, Hamed Dashti¹, Hamid R. Rabiee¹,²,*

¹ Al-Med Group, AI Innovation Center, Sharif University of Technology, Tehran, Iran
² DML Lab, Department of Computer Engineering, Sharif University of Technology, Tehran, Iran
³ Department of Radiology, Tehran University of Medical Sciences, Tehran, Iran
⁴ Preventive Medicine and Public Health Research Center, Psychosocial Health Research Institute, Community and Family Medicine Department, School of Medicine, Iran University of Medical Sciences, Tehran, Iran
⁵ Cardiovascular Medical & Research Center, Iran University of Medical Sciences, Tehran, Iran
⁶ Isfahan University of Medical Sciences, Isfahan, Iran
⁷ Kerman University of Medical Sciences, Kerman, Iran
⁸ Research Institute of Animal Embryo Technology, Shahrekord University, Shahrekord, Iran
⁹ Kashan University of Medical Sciences, Kashan, Iran
¹⁰ Department of Radiology, Kashan University of Medical Sciences, Kashan, Iran
¹¹ Department of Pediatrics, Kashan University of Medical Sciences, Kashan, Iran
¹² Department of Computer Engineering, Imam Khomeini International University, Qazvin, Iran
¹³ Shaheed Beheshti University of Medical Sciences, Medical Academy of Science, Tehran, Iran

* To whom correspondence should be addressed: Hamid R. Rabiee: rabiee@sharif.edu,
Abstract

**Background:** Since late 2019, COVID-19 has been spreading all over the world and caused the death of many individuals. The virus has a very high transmission rate, and therefore, demands the rapid identification of infected patients to reduce the spread of COVID-19 among people. The Reverse-Transcription Polymerase Chain Reaction (RT-PCR) has been the gold standard test for detecting the virus, however, it suffers from a high rate of false-negatives. CT-scan imaging has been used to detect COVID-19 in patients as an alternative method which offers much higher accuracy and sensitivity. To diagnose COVID-19 among other viral pneumonia and asymptomatic patients have proven to be challenging tasks, and requires experienced radiologists to perform the diagnosis. However, when the pandemic reaches a peak, hospitals are overwhelmed with patients and experience shortage of medical staff, meaning that expert radiologists may not have the capacity to preform accurate and rapid diagnosis of COVID-19 for all patients in a timely manner. In those situations, artificial intelligence (AI) based diagnosis systems can help the radiologists and physicians to accelerate the process of diagnosis, increase the accuracy, and predict the severity of the disease. An AI-based system must provide high accuracy and sensitivity on large and diverse test datasets, be generalizable to unknown CT-scan devices, and show no bias to the batch effect of special CT-scan devices. It should also be interpretable to be acceptable by radiologists and physicians. The previous AI-based diagnosis systems have not been able to address all of the points motioned above.

**Method:** We designed an interpretable deep neural network to diagnose patients with COVID-19 and patients with other pneumonia diseases from axial lung CT-scan images. Our network also detects the infected areas of the lung and is able to calculate the percentage of infection. In the preprocessing step, we separated the area related to the lung lobes and to avoid hospital and CT-scan device-related batch effects that were most obvious in the lung peripheral and background of images. We adopted a weakly supervised method to train our deep model without having any label for infected parts and any tags for the slices of the CT-scan images that had signs of disease.

**Results:** We collected a total of 4794 CT-scan images with clinical evidence, including 3652 CT-scans from patients at different stages of the COVID-19 infection, 572 CT-scans from healthy patients, and 570 CT-scans from patients with non-COVID-19 lung infections. The CT-scans were collected from 12 different hospitals and radiology centers across different cities in Iran. We used samples from 6 medical centers for the training and evaluation of the model, and reserved samples from the remaining 6 centers to evaluate the generalizability of the model on unseen cases. Our trained model achieved an average accuracy of 96.56% when detecting infected samples against the healthy ones for test samples from the 6 medical centers used in the training phase (resulting in the sensitivity of 97.25% for COVID-19, the sensitivity of 100% for other diseases, and specificity of 87.50%). In addition, the model achieved an average accuracy of 96.12% on the COVID-19 samples from the unseen medical centers. It also achieved an accuracy of 95.71% in distinguishing COVID-19 samples from other diseases (sensitivity of 98.15% and specificity of 81.03%) for the test samples collected from known centers, and accuracy of 95.73% on the COVID-19 samples collected from the unseen centers. Moreover, some of the Covid-19 samples in the test data had no visible sign of infection to the naked eye, however our proposed model was able to accurately detect infected cases. In a live session at Tehran Radiology Center, the proposed model successfully detected all the patients infected with COVID-19 and other diseases. It is important to note that we only had healthy samples from the same center while training our model. We also received a report from Imam Ali Hospital in Maragheh, which had extensively used our system for several months, reported that COVID-19, healthy, and other diseases were detected with very high accuracy and sensitivity.
**Conclusions:** In this manuscript, we propose a new interpretable deep learning model that was trained with a unique weakly supervised method. It is capable of detecting healthy patients, patients with COVID-19, and patients with other diseases with high accuracy and high sensitivity using 47794 test datasets of high diversity. It is important to note that the model is interpretable by architecture which means it can highlight the areas that are significant in detection of COVID-19 patients. In addition, the proposed algorithm can specify the infected areas of lung and calculate the percentage of infected lung volume. These characteristics would greatly help radiologists and physicians to implement precise treatment for each patient.

The live testing of the trained model on diseased samples from a hospital where we only had healthy samples in the training phase, proves that the model was not biased toward hospital and device-related features of the CT-scan images and can successfully reduce the unwanted batch effects. It has also proved to identify COVID-19 patients that had no visible sign of infection in their CT-scan images at the early stages of infection. Finally, its high performance on previously unseen samples from different hospitals and devices confirms the generalizability of the proposed algorithm.

**Introduction**

The novel coronavirus was first identified in China towards the end of 2019, and the World Health Organization (WHO) referred to the virus as COVID-19 [1]. The virus has a high rate of transmission [2] which has terrified people across the world. Given that no confirmed treatment or vaccine has been developed for COVID-19 thus far, an accurate and fast diagnosis is vital to reduce the speed of transmission [3-5].

Although the COVID-19 is typically diagnosed using reverse-transcription polymerase chain reaction (RT-PCR) [6, 7] as the reference standard for diagnosis of positive infection of coronavirus, the test results in many false negatives, and several studies have shown that chest CT-scans can be used as a more accurate alternative for the diagnosis of COVID-19 with a low rate of false negatives and above 90% accuracy [8-10]. According to RSNA, some of the COVID-19 patients show no sign of infection in their CT-scan during the first two days of infection, and the anomalies of their CT-scans are not visible [11, 12]. Moreover, a high number of patients visit hospitals to be diagnosed when the spread of the virus is at its peak, which overwhelms the limited number of expert medical staff that can attend to the diagnosis of all patients in a timely manner. An Artificial Intelligence (AI) system can help to accelerate the diagnosis process by aiding the prioritization of high risk cases, and to increase the diagnosis speed and accuracy. The distinction between COVID-19 pneumonia and other lung-related diseases is another challenge for inexperienced radiologists and medical staff that reduce the accuracy of the diagnosis and treatment processes.

Previous studies have demonstrated that the use of deep networks and artificial intelligence methods can enhance how we tackle the challenges related to COVID-19. A study from China, collected 4356 chest CT-scans from 3,322 patients across 6 hospitals. They developed a model named COVNet that generates a probability score for COVID-19, community-acquired pneumonia (CAP), and non-pneumonia samples. They achieved 90% sensitivity for COVID-19 patients, 87% for CAP, and 94% for non-pneumonia [13]. In another study, Wang and Wong designed an architecture called COVID-Net and trained it with 13,975 Chest X-Ray (CXR) images instead of CT-scan images with 91%, 94%, and 95% sensitivities for COVID-19, non-COVID-19, and normal cases, respectively [14]. In an initial study, Gozes et. al. used multiple international datasets, including disease-infected areas in China. They used ResNet50 which was pre-trained on ImageNet and achieved an Area Under the Curve (AUC) of 0.994 with 94% sensitivity and 98%
specificity. They also used results collected from 56 COVID-19 patients, and 51 non-COVID-19 patients and achieved a sensitivity of 96.4% and specificity of 98% [15]. Shah et al. used 738 images from public datasets [16] and designed a new architecture named CTnet-10 that achieved an accuracy of 82.1%. They also used a pre-trained VGG-19 and transfer learning which resulted in 91.78 % accuracy [17]. In a different study, Wang et al. trained a DNN on 453 images of pathogen-confirmed COVID-19 cases. They achieved a total accuracy of 73% with a specificity of 67% and a sensitivity of 74% [18]. Finally, Xu et al. used 618 CT samples to train a location attention model that separates influenza-A viral pneumonia, COVID-19, and locations that are irrelevant to infection. This location attention model yielded an overall accuracy of 79.4% [19].

Despite the success of recent deep learning methods in producing accurate diagnosis of COVID-19 using CT-scan images with high sensitivity, they lack interpretability. They also do not explicitly address the batch effect removal issues, and thus do not generalize well on images from different sources. Moreover, they mostly utilize supervised machine learning algorithms for training and detection, while supervised methods require a great effort to tag all slices of a large number of CT-scan images. Due to these shortcomings, we propose a new accurate and rapid diagnosis assistant system by utilizing statistical and deep machine learning methods in AI to distinguish patients with COVID-19 from healthy patients and the ones that had diseases other than COVID-19 in an interpretable way, while removing the batch effects. Moreover, the system we designed can highlight the involved regions and calculate the percentage and location of the infected lung volume. We trained our system on a big cohort of CT-scan images collected from different hospitals across different cities in Iran, in order to support CT-scan images with different characteristics.

Method

The general workflow for the proposed interpretable COVID-19 detection is shown in Figure 1. In the first step of pipeline, the Ground Glass Opacity Axial (GGOA) CT-scan images are preprocessed and the lobes of lungs are detected and extracted from the axial slices. The images of the left and right lobes of all the slices are then fed into two deep Convolutional Neural Networks (CNNs), one for calculating the probability of being diseased versus healthy, and the other for calculating the probability of diagnosis to be COVID-19 versus other diseases. In addition to the probability, the first network also detects the infected areas in lung images. In addition, both networks specify the effective areas of the lungs that the decision is made based on them. In the final step, the probabilities assigned to the lobes are aggregated to make a final decision for the whole sample. Each part of the pipeline is explained in detail in the preceding subsections.

Preprocessing of CT-scan images

The GGOA CT-scan images are stored in DICOM files. The value calculated by the CT-scan device for each voxel, contains the scanner-dependent information for converting the scanned values into the Hounsfield units. This transformation is linear and requires the values for slope and intercept that are stored in the header of each DICOM file. The values in the Hounsfield unit should range between -2000 to 3000. Since the values out of this range correspond to artifacts, the values lower than -2000 were set to -2000, and the values above 3000 were set to 3000. Thereafter, the minimum value was subtracted from the image to make all the values positive. There were also differences in the background of the images captured by different CT-scan devices. To make all images normalized into the range [0, 2048], the median value of the background in all slices of one CT-scan image was calculated and subtracted from all the values (the negative values were set to 0). The values above 2048 were then clipped into 2048, as they were related to the details in the bones which are irrelevant to the objective of this study. All the values were transformed into a
range of [0, 1] by dividing them by 2048. In the final step, a median filter with a kernel size of 3 was applied to eliminate nonlinear noises from the images.

**Figure 1** - The general workflow for interpretable COVID-19 detection.

**Detecting lobes**

Each lung image has two similar groups of pixels, those related to soft tissues and bones, and those related to the lung and air. These two groups were separated by applying the K-means clustering algorithm to all pixels in each image. The pixels assigned to the group with the lower value of the center were labeled as the foreground, as the lung and air fall in this group. In the next step, the foreground was eroded by a square kernel of size 3 to eliminate the small noisy isolated areas. The remaining foreground was then dilated by a square kernel of size 8, so the disjoint close areas become connected. Each group of connected pixels was then separated as a single object. The objects with an area less than 5000 pixels, which correspond to small holes, were eliminated. Apart from the background of the lung which expands everywhere, the larger object on the left side of the image was selected as the left lobe, and the larger object on the right side of the image was selected as the right lobe. The whole image inside a rectangular bounding box around each
lobe was cropped and copied into the center part of a 256 by 256 image with a black background. In most cases, the cropped image was smaller than the mentioned size. In exceptional cases where the cropped image was bigger than 256 in at least one of the dimensions, it was scaled to fit in a 256 by 256 image for keeping the width to height ratio and then copied into the center of the black background.

**Infected versus healthy decision maker**

The structure of the proposed deep network that is being used for detecting the virus in the lobes is presented in Figure 2. The network receives the images related to one lobe in three consecutive slices as a 256 by 256 by 3 array as input and calculates the features related to the middle slice. The images related to the previous and next slices are used to provide extra information about the continuity of white material of the middle slice. The input is fed into a convolutional subnetwork (Table 1) with an output size of 32 by 32 by 256 which is related to 32 by 32 mesh of neurons with 256 features for each. The receptive field of each neuron in this network is a 36 by 36 patch in the input image so that the extracted features for each neuron are related to the 36 by 36 patch in the input image.

Making a decision about one patch is not possible by solely using the features extracted from that patch. Therefore, we need to look at a bigger area around the patch in order to predict its state. For example, if a patch is solid white, we don’t know whether the patch is related to the lung periphery or an infected part inside the lung unless we look at its neighboring patches. To add extra information about the continuity of white material of the middle slice. The network is fed into a U-Net-style [20] encoder-decoder (Table 1) so the features are extracted from a bigger receptive field in the encoder part, and the resolution is increased to the input size in the decoder part. In addition to these sets of features, the location of each patch and its distance from the lung peripheral plays an important role in distinguishing COVID-19 from other diseases, as COVID-19 related infections are observed to start from the peripheral. The Manhattan distance between each pixel and the closest peripheral pixel is calculated by using the Breadth-First Search (BFS) algorithm, and the minimum distance of the pixels in each patch is assigned to that patch. The features extracted for each patch, the ones extracted for the vicinity of the patch, and the minimum distance of the patch from the lung peripheral are concatenated to form the final features for each patch.

Features set of each patch is fed separately into a fully connected subnetwork (Table 1) that calculates the probability of the patch to be infected. Another U-Net style decoder has been used to separate infections in finer details of 6x6 patches from the 36x36 patches that have been identified as infectious. The details of network architecture are explained in Table 1. The two-level identification of infections helps in decreasing a large number of false positives by forcing the model to concentrate in larger areas around smaller 6x6 patches.

The features of all patches are also fed into a fully connected subnetwork (Table 1) to calculate a bounded attention weight between 0 and 1 for each patch. The features of patches are multiplied by the calculated weight and added up to calculate the final features for the lobe. This network works only based on patches so the final decision can easily be mapped into the responsible patches. This is a logical scheme for detecting various diseases in the lung as if the network finds any evidence of diseases in a limited number of patches, it will assign it to the entire lobe and in the case of COVID-19, ground-glass opacity infections near the periphery of the lung has been proven to be one of the main pieces of evidence.

**COVID-19 versus other diseases decision maker**

Another deep network is adopted to distinguish COVID-19 from other diseases. This network also has the same structure as the previous one, except it does not have the part related to detecting infection in the patches. The base part for extracting features for the patches could have been
shared between these two networks, but in practice, learning different features for the patches by two separate networks, resulted in better performance.

**Figure 2** - The deep model used for calculating the probability of each slice lobe to be diseased and producing infection map for it.

**Table 1** - Specification of the layers used in each subnetwork of the model used for diseased versus healthy detection

| Subnetwork                        | Layers                                                                 |
|-----------------------------------|------------------------------------------------------------------------|
| Patch feature extractor           | conv(64x3x3,s=1), relu, conv(64x3x3,s=1), relu, MaxPool(2x2,s=2), [Name of the output: PFE1]  
  conv(128x3x3,s=1), relu, conv(128x3x3,s=1), relu, MaxPool(2x2,s=2) [Name of the output: PFE2]  
  conv(256x3x3,s=1), relu, conv(256x3x3,s=1), relu, MaxPool(2x2,s=2) |
| Vicinity feature extractor (Encoder) | conv(64x1x1,s=1) [Name of the output: VFE1]  
  conv(64x3x3,s=2), conv(64x3x3,s=1), relu [Name of the output: VFE2]  
  conv(64x3x3,s=2), conv(64x3x3,s=1), relu |
| Vicinity feature extractor (Decoder) | TransposeConv(64x2x2,s=2), conv(64x3x3,s=1), relu, UNet concatenation with VFE2  
  TransposeConv(64x2x2,s=2), conv(64x3x3,s=1), relu, UNet concatenation with VFE1 |
| Patch infection label decider                          | Conv(64x1x1,s=1), relu  
|                                                   | Dropout(0.5),  
|                                                   | Conv(2x1x1,s=1), Softmax |
| Fine infection label decider                       | Conv(64x1x1,s=1),  
|                                                   | TransposeConv(64x2x2,s=2), conv(64x3x3,s=1), relu  
|                                                   | UNet concatenation with PFE2,  
|                                                   | TransposeConv(64x2x2,s=2), conv(64x3x3,s=1), relu,  
|                                                   | UNet concatenation with PFE1,  
|                                                   | Dropout(0.5), Conv(64x1x1,s=1), relu, Dropout(0.5), Conv(2x1x1,s=1), Softmax |
| Bounded attention weights calculator               | Conv(64x1x1,s=1), relu  
|                                                   | Dropout(0.5)  
|                                                   | Conv(2x1x1,s=1), Softmax |
| Lobe label decider                                 | Linear(64), relu  
|                                                   | Dropout(0.5)  
|                                                   | Linear(1), Softmax |

**Aggregating the results of the lobes**

All the lobes from a sample are fed into the aforementioned networks and the probabilities of being diseased versus healthy, and for the disease to be COVID-19 as opposed to other diseases are calculated for each of them. As the signs of diseases should be continuous in the height of the lung, we expect to have at least two consecutive diseased lobes to consider the sample as diseased. Therefore, the probability of each slice lobe to be diseased is softened by the probability of the same lobe in its adjacent slices. Here, we used the minimum between the probability assigned to the same lobe and the maximum probability assigned to the two adjacent lobes as the softened probability for the lobe. Finally, a CT-scan’s probability to be diseased is calculated by the maximum probability of its slices lobes being diseased. This process eliminates many of the false positives of decision making-based on a single slice lobe. The same procedure is also repeated in calculating the probability of a CT-scan to be related to COVID-19 versus other diseases.

**Training the model**

It is typical to use the cross-entropy loss in classification problems. However, in our problem, we have the challenge of having many slices in each sample, and hence the final decision is made based on the aggregation. Using operations such as maximum in the training phase requires the presence of all images of the slice lobes related to one CT-scan to be in the training batch. Nevertheless, this is not possible in ordinary GPU RAMs as each CT-scan has about 100 slices or 200 lobes, and we also need to have batches of different samples to have more efficient optimization steps. Moreover, this makes the training process time consuming as we need to run a large network over all the images, for every sample. It also has the problem of biasing the gradient flow towards only one or a small number of images. This makes the progress in training slow, and the network needs to be trained for much more epochs to learn something meaningful. It may also result in learning the evidence of the highly infected slices, and lose the power of generalization in slightly infected CT-scans. One solution is to train the network to label the slice lobe labels instead of the entire CT-scan, but the ground truth labels are not available. We also cannot assign the CT-scan labels to all of their slice lobes as the virus may have affected a part of the lung only. Furthermore, for training the patch labeling part, we require the infected area to be specified in all slices. This is a laborious and time-consuming effort for radiologists, especially at these peak times.

---

| Patch infection label decider                          | Conv(64x1x1,s=1), relu  
|                                                   | Dropout(0.5),  
|                                                   | Conv(2x1x1,s=1), Softmax |
| Fine infection label decider                       | Conv(64x1x1,s=1),  
|                                                   | TransposeConv(64x2x2,s=2), conv(64x3x3,s=1), relu  
|                                                   | UNet concatenation with PFE2,  
|                                                   | TransposeConv(64x2x2,s=2), conv(64x3x3,s=1), relu,  
|                                                   | UNet concatenation with PFE1,  
|                                                   | Dropout(0.5), Conv(64x1x1,s=1), relu, Dropout(0.5), Conv(2x1x1,s=1), Softmax |
| Bounded attention weights calculator               | Conv(64x1x1,s=1), relu  
|                                                   | Dropout(0.5)  
|                                                   | Conv(2x1x1,s=1), Softmax |
| Lobe label decider                                 | Linear(64), relu  
|                                                   | Dropout(0.5)  
|                                                   | Linear(1), Softmax |
when thousands of patients are waiting to be examined. Therefore, it is necessary to utilize a weakly supervised\(^2\) procedure for finding infected slices and infected patches.

**Weakly supervised training for labeling the samples**

To overcome the problem of unknown lobe labels, a set of 10 nearly equally distanced slice lobes are selected across the height of each sample in the batch. Therefore, the images will somehow cover different parts of the lung and we can expect that the virus would be observable in at least some of them. The label of each sample is assigned to 30% of the top infected slice lobes, which in this case would be 4, as the number is rounded up, and the cross-entropy loss is calculated only for these top slice lobes. To keep the balance, the top 30% infected slice lobes are included in loss calculation of healthy samples. The same procedure is also repeated for training the network for detecting COVID-19 vs. other diseases.

**Using fake labels for weakly supervised training of the infections**

To deal with the problem of unknown infection masks for training the patch infection detection module, a semi-infection mask is calculated for each slice lobe. In fact, we highlight some areas that might be related to infection in the diseased samples, while all the patches of healthy samples are assumed to have no infection. Therefore, we expect the network to distinguish between the patterns similar to infection that only occur in diseased samples and the ones related to other phenomenon (e.g. appearance of the top of the liver, which is common in healthy and diseased samples). In other words, we highlight the pixels with a color between the lung peripheral and background which would include infectious parts and other parts (e.g. faded lung peripherals) that are not eliminated by the thresholding process. We use fake infection labels for these areas but in the healthy samples, all of the patches are assumed to be healthy. Therefore, the model learns to distinguish them.

The suspected areas for infection are the ones that are neither absolute black nor too white, such as soft tissues and bones. To ensure that a pixel is not related to the background or lung we simply need to use a threshold greater than the standard deviation of all the images in the subject’s CT-Scan. In this way, assuming all the images have nearly the same distribution of colors, we can be sure to detect these areas with a confidence level of 95%. For finding the upper bound that excludes pixels that are too white to be related to an infection (tissues and bones), we use the distribution of values in all the images in the subject’s CT-Scan. We expect to see two peaks in the distribution plot, the first one related to the black sides and the second one related to the white sides. We select the value with the minimum intensity between these two and use it as the threshold to separate them. To assign these pixel-wise infections to the patches of 36x36, a threshold of $\frac{1}{3}$ of density is used.

Training the infection detection subnetwork is conducted using cross-entropy loss between the predicted probability and the assigned semi-infection label for the diseased samples and healthy labels for all the healthy samples. We must note that the proposed labels are not ground truth and might be wrong. Among these labels, there might be proposed infected labels that are not related to the infection, as well as some missed infected parts. To deal with these wrong labels, we set the concordance score, the distance between the label and the probability assigned, to 1. Then for the top 80% patches, the loss is calculated by a factor of 1 and for the rest, it is calculated with a factor of 0.1, so the network can ignore them more easily. As there are much more non-infected patches than the infected ones, the loss related to the infected patches is weighted as the total factor related

\(^2\) Weak supervision is from the perspective that we have sample-level labels but do not have more precise labels for slices and infected zones and they are the ones that are actually needed.
to non-infected patches in loss divided by the total factor of infected patches. Therefore, the loss injected into the network by both infected and non-infected patches would be balanced.

The fine infection labeler network was almost trained in the same manner, with some differences in details. Any 6x6 patch in which more than ¾ of its pixels were labeled as a probable infection, and at least one of the coarser 36x36 patches covering it was identified as infectious, with a probability above 0.5, is labeled as infected. The same weighting scheme was adopted to balance the total effect received by the network from both the non-infected and infected patches.

**Concordance loss to coordinate the labels of slices and the infection labels of their patches**

For COVID-19, we expect to see the infection in the lobes detected as diseased. Although the patch-labeling part and the lobe-labeling part in the proposed model work on the same set of features, their infection status is decided independently. For example, the lobe-decider may assign the diseased label for one lobe while there is no infected patch with the probability of more than 50%. To keep the concordance and to make the network more interpretable, an MSE loss is also added between the probability of being diseased assigned by the lobe-labeler and the probability of the most infected coarse patch (36x36). We only add this loss term for healthy and COVID-19 samples as the other diseases might not include infection. This makes the network to bold out at least one infected patch for the lobes with high COVID-19 probability. It also helps the network to identify diseased lobes more easily.

**Parameters of training**

For training the network, we used the total loss which includes the lobes labels loss and the concordance loss with a weighting factor of 1, and the patches labels loss with a weighting factor of 0.01. 10% of the dataset was used as the test data, 9% as the validation set to help in hyperparameter selection, and the remaining 81% was used to train the model. The model, except for the fine infection labeler subnetwork, were trained using Adam optimization algorithm [21] with default parameters, and an initial learning rate of 1e-4. Batches of 6 samples with 10 slices for each were chosen to train the network on an Nvidia GeForce 2080 Ti GPU with 11 Gb of RAM. The network was trained for 500 epochs and the epoch related to the highest validation accuracy was chosen as the final model. To have a better initial point, the subnetwork for extracting features for the patches was initialized with a VGG-16 network [22] trained on ImageNet [23] available in the torchvision package [24]. After this step of training, the entire network was freezeed and only the fine infection labeler network was trained using Adam optimization algorithm with an initial learning rate of 1e-5 using batches of 14 samples with two slices for each sample on a GPU with 6Gb of RAM.

**Calculating the percentage of the infected volume of the lung**

As explained in the previous subsections, we detect the lobes of the slices, feed them to a deep model, and the model returns the probability of the patches of each slice that contain the COVID-19 infection. The probability of being infected is calculated for two levels of patches; a coarser infection map for 36x36 patches with an overlap of 28 pixels in each dimension, and a finer infection map for 6x6 patches with an overlap of 4 pixels. For each map, the probability of the patch being infected is assigned to all of its pixels, and for the pixels that are included in multiple patches, the maximum probability is considered for those pixels. For each map, the pixels having a probability of more than 0.5 are assumed to be infected. The two maps are multiplied with each other to form the final infection map. As a result, the coarser map detects infections in a more robust manner, and the finer map separates the infected area with a higher resolution from the coarse map. The total volume of the infected area is calculated by adding the number of the pixels identified as infected, and the total volume of the lung is calculated by counting the number of the
pixels related to the lung lobes in all of the slices which were detected in the preprocessing phase. The ratio of these two numbers determines the percentage of the lung volume that is infected.

**Interpreting the decisions made by the model**

As explained in the architecture of the model, the model labels each slice based on an attention mechanism over the features calculated for all 36x36 patches of a slice. In this way, the patches having the marks of diseases that have caused the model to make its decision can be found easily, and verified visually. Moreover, the concordance loss that was defined for making the probability of each slice to have signs of COVID-19 and the probability of its most probable infected coarse patch closer together, would bold out the infected areas, especially the ones that are less visible with the naked eye, at the first glance.

**Results**

We collected 4794 CT-scan samples from Tehran Radiology Center (TRC), Imam Khomeini, Yaas, Rasoul Akram, Firoozgar, and Amiralam Hospitals in Tehran, Afzalipour and Bahonar Hospitals in Kerman, Imam Khomeini Hospital in Qazvin, Isa Ibn Maryam, Amin and Goldis Hospitals in Isfahan, and Shahid Beheshti Hospital in Kashan. The diversity of CT-scan samples has helped us to effectively eliminate batch effects and optimize our system performance to support images from different CT-scan imaging devices with various properties and cut thicknesses.

The entire dataset contained 3652 CT-scans from COVID-19 patients approved by RT-PCR tests, 572 healthy CT-scans (including CT-scans from the years before COVID-19 was identified), and 570 CT-scans related to other diseases with similar symptoms as COVID-19. The samples were chosen from patients at different stages of COVID-19, including patients at the initial stage of infection with no visible sign in their CT-scan to the ones in the consolidation stage. The details of the samples are presented in Table 2. The samples from Isa Ibn Maryam, Amin, Goldis, Rasoul Akram, Firoozgar, and Amiralam’s Hospitals were not used in the training phase. Instead, they were utilized to evaluate the performance of the model on unseen CT-scans collected from different hospitals. The remaining CT-scan samples were divided into training, validation, and test sets with a ratio of 0.81, 0.09, and 0.1, respectively.

The model for detecting infected versus healthy samples achieved an accuracy of 96.56% (sensitivity of 97.25% for COVID-19, sensitivity of 100% for other diseases, and specificity of 87.50% for the healthy samples) over the test data of the seen hospitals, and an average accuracy of 95.8% for the unseen hospitals. The model for distinguishing COVID-19 from the other diseases achieved an accuracy of 96.12% (sensitivity of 98.15% and specificity of 81.03%) over the test data of the seen hospitals, and 95.73% for the unseen hospitals. The evaluation metrics calculated for the test data of the hospitals included in the training phase and the unseen hospitals for distinguishing diseased samples from the healthy and COVID-19 samples from the other diseases are presented in Table 3. The evaluation metrics for distinguishing COVID-19 from other diseases are presented with details in Table 4.

| Hospital/Center          | Total | COVID-19 Patients | Non-COVID-19 Patients | Healthy | Status         |
|--------------------------|-------|-------------------|-----------------------|---------|----------------|
| Imam Khomeini (Tehran)   | 188   | 188               | 0                     | 0       | Used in training |
| Yaas (Tehran)            | 38    | 38                | 0                     | 0       | Used in training |
| TRC (Tehran)             | 438   | 0                 | 0                     | 438     | Used in training |
| Afzalipour/Bahonar (Kerman) | 513   | 103               | 276                   | 134     | Used in training |
Table 3 - Evaluation metrics for the test data from different hospitals for distinguishing diseased samples from healthy samples

| Hospital/Center | Accuracy | Sensitivity (COVID-19) | Sensitivity (Other diseases) | Specificity (Healthy) |
|-----------------|----------|------------------------|------------------------------|-----------------------|
| Imam            | 95.24    | 95.24                  |                              |                       |
| Yaas            | 100      | 100                    |                              |                       |
| TRC             | 95.35    |                        |                              | 95.35                 |
| Kerman          | 91.38    | 100                    | 100                          | 64.29                 |
| Qazvin          | 98.21    | 96.15                  | 100                          |                       |
| Kashan          | 97.02    | 97.02                  |                              |                       |
| Isfahan (unseen)| 93.26    | 93.26                  |                              |                       |
| Firoozgar (unseen)| 96.07  | 96.07                  |                              |                       |
| Rasoul (unseen) | 95.71    | 95.71                  |                              |                       |
| Amiralam (unseen)| 96.00  | 96.00                  |                              |                       |
| Total (Seen)    | 96.56    | 97.25                  | 100                          | 87.50                 |
| Total (Unseen)  | 95.8     | 95.8                   |                              |                       |

Table 4 - Evaluation metrics for the test data from different hospitals for distinguishing COVID-19 patients from other diseases

| Hospital/Center | Accuracy | Sensitivity (COVID-19) | Specificity (Other diseases) |
|-----------------|----------|------------------------|------------------------------|
| Imam            | 100      | 100                    |                              |
| Yaas            | 100      | 100                    |                              |
| TRC             | 87.18    | 90.91                  | 85.71                        |
| Kerman          | 76.79    | 76.92                  | 76.67                        |
| Qazvin          | 100      | 100                    |                              |
| Kashan          | 88.76    | 88.76                  |                              |
| Isfahan (unseen)| 98.65    | 98.65                  |                              |
| Firoozgar (unseen)| 89.14  | 89.14                  |                              |
| Rasoul (unseen) | 100      | 100                    |                              |
| Amiralam (unseen)| 96.12  | 98.15                  | 81.03                        |
| Total (Seen)    | 95.73    | 95.73                  |                              |
| Total (Unseen)  | 95.73    | 95.73                  |                              |

The performance of the model was also evaluated by radiologists in a live session at Tehran Radiology Center (TRC), from which we had only received healthy samples. The model labeled all the 6 random samples correctly (1 healthy, 2 COVID-19, and 3 non-COVID diseased samples). Detecting diseased samples while the model was only trained with healthy samples from this center, proves that the model is not biased towards the center’s CT-scan device. The model also detected all the infected areas in the diseased samples, correctly. In addition, we received a report from Imam Ali Hospital in Maragheh that had voluntarily used our system, stating that our algorithm correctly diagnosed COVID-19, healthy, and other diseased samples. An example of the system output is presented in Figure 3.
The features of all patches are also fed into a fully connected subnetwork (Table 1) to calculate a bounded attention weight between 0 and 1 for each patch. The features of patches are multiplied by the calculated weight and added up to calculate the final features for the lobe. This network works only based on patches, so the final decision can easily be mapped into the responsible patches. This is a logical scheme for detecting various diseases in the lung as if the network finds any evidence of the disease in a limited number of patches, it will assign it to the entire lobe and in the case of COVID-19, ground-glass opacity infections near the periphery of the lung has been proved to be one of the main pieces of evidence in predicting the labels of the lobes.

**Figure 3** - A sample of the proposed system output. The infected part has been captured and specified with a red box, and the system diagnosis results along with the infected volume are shown on the upper right corner.

Our system has also been able to detect COVID-19 from the CT-Scan images with no visual signs of infection. Two such samples are shown in Figure 4. We have observed that the system captures a similar pattern in the lower parts of the lung which may be a sign that the COVID-19 virus starts from parts closer to the spinal cord and then spreads to the other parts of the lung.
Figure 4 - Two samples with positive RT-PCR that have no obvious visual sign of infection in their CT-Scans. Our system captures a similar pattern in the peripheral sides of the lungs that are near to the spinal cord in most of them.

Discussion

In this manuscript, we proposed an interpretable deep learning model for rapid and accurate diagnosis of CT-scan images of lungs to distinguish healthy patients, patients with COVID-19, and patients with other diseases. To address the problem of batch effect in different CT-scan devices and hospitals that is most obvious in the peripheral parts of the lungs, and the background of the images, we separated the lobes from the images and used them as the input to the model. We proposed a unique weakly supervised method for training the model based on the labels of the samples only, without requiring detailed annotations of the slices containing the signs of diseases and the infected area, as it took a great effort to tag all the slices of CT-scan images. Therefore, we were able to use a large cohort for training the model, and hence, making it more generalizable. We gathered 4794 CT-scan images from 12 different hospitals and medical imaging centers of different cities to cover different CT-scan devices, cut thicknesses, and radiation dosages which contained CT-scan images of different stages of COVID-19 along with images from healthy people and patients with pneumonia diseases other than COVID-19 with infections in their lungs. We used samples from half of the centers to train and evaluate the model and another half for evaluating the generalizability of the model on unseen images from different devices. Our trained model reached an accuracy, sensitivity and specificity similar to the top metrics reported in other researches in distinguishing COVID-19 patients from healthy people and patients with other diseases on a large number of test samples. Reaching a performance similar or better than the models trained in a supervised scheme with fully annotated data shows the efficiency of our unique weakly supervised training method. Furthermore, the model reached a similar accuracy using 1435 samples from the 6 centers not used in the training process, which proves its generalizability. The model also succeeded in detecting diseased and COVID-19 CT-scans from a center used in the training phase from which we only had healthy samples. This proves that the model is not biased toward the batch effect of images related to a special CT-scan device or a specific imaging center. Our model passed the tests successfully in a number of live sessions. We also proved that our model could detect COVID-19 in patients without observable signs of infection in their CT-scans, in the early stages of the disease.

The high performance, sensitivity, generalizability, and unbiasedness of the proposed model makes it suitable to be used as an AI assistant at medical centers, especially when the pandemic reaches a peak, hospitals are overwhelmed with patients and experience shortage of experts, meaning that expert radiologists may not have the capacity to preform accurate and rapid diagnosis of COVID-19 for all patients.
The system can provide an instant diagnosis on the state of the patients, so they can be quarantined as soon as possible, in order to avoid the spread of virus. The model is also able to label all the slices and show the worst slices to the experienced radiologists for effective diagnosis of the disease. As mentioned before, the model can specify the infected areas of the lung and calculate the total size of the lung so it can calculate the percentage of the infected volume of the lung which can provide a measure for severity of the virus in patients. Therefore, it can be used as a prioritization tool at peak times in order to offer treatment to patients with severe conditions with higher priority. Since the system does not utilize a black-box deep learning model, it can highlight the area of the image that are affected by the virus. This can help radiologists to understand whether the model makes its decisions logically, and use that information to adopt an optimal strategy for treatment of patients.
Author Contributions

HD and HRR designed the experiment. RG, MH, SM, HD, HL, and HRR designed the model and the training method. SM, MH, and HL designed the statistical pre-processes. RG and MH trained the model. RG, MH, SM, AM, SH, FAN, AE, MSK, RE, JA, AA, MRS, AK, AF prepared the data, helped the experimental setup and advised the study from the medical point of view. All the authors helped in writing the manuscript.

Funding

This work was supported in part by Iran National Science Foundation (INSF) Grant No. 96006077 and ISTI grant number 11/41701.

Reference

1. WHO homepage, https://www.who.int/. [visited 2020/05/05].
2. Li, Q., et al., Early transmission dynamics in Wuhan, China, of novel coronavirus–infected pneumonia. New England Journal of Medicine, 2020.
3. Huang, C., et al., Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. The lancet. 2020. 395(10223): p. 497-506, 0140-6736.
4. Zhu, N., et al., A novel coronavirus from patients with pneumonia in China, 2019. New England Journal of Medicine, 2020.
5. Hui, D.S., et al., The continuing 2019-nCoV epidemic threat of novel coronaviruses to global health—The latest 2019 novel coronavirus outbreak in Wuhan, China. International Journal of Infectious Diseases, 2020. 91: p. 264-266, 1201-9712.
6. Zu, Z.Y., et al., Coronavirus disease 2019 (COVID-19): a perspective from China. Radiology, 2020: p. 200490, 0033-8419.
7. Chu, D.K.W., et al., Molecular diagnosis of a novel coronavirus (2019-nCoV) causing an outbreak of pneumonia. Clinical chemistry, 2020. 66(4): p. 549-555, 0009-9147.
8. Ai, T., et al., Correlation of chest CT and RT-PCR testing in coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases. Radiology, 2020: p. 200642, 0033-8419.
9. Li, Y. and L. Xia, Coronavirus disease 2019 (COVID-19): role of chest CT in diagnosis and management. American Journal of Roentgenology, 2020. 214(6): p. 1280-1286, 0361-803X.
10. Lei, J., et al., CT imaging of the 2019 novel coronavirus (2019-nCoV) pneumonia. Radiology, 2020. 295(1): p. 18-18, 0033-8419.
11. Simpson, S., et al., Radiological Society of North America Expert Consensus Statement on Reporting Chest CT Findings Related to COVID-19. Endorsed by the Society of Thoracic Radiology, the American College of Radiology, and RSNA. Radiology: Cardiothoracic Imaging, 2020. 2(2): p. e200152, 2638-6135.
12. Shi, H., X. Han, and C. Zheng, Evolution of CT manifestations in a patient recovered from 2019 novel coronavirus (2019-nCoV) pneumonia in Wuhan, China. Radiology, 2020. 295(1): p. 20-20, 0033-8419.
13. Li, L., et al., Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT. Radiology, 2020.
14. Wang, L., Z.Q. Lin, and A. Wong, Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. Scientific Reports, 2020. 10(1): p. 1-12, 2045-2322.
15. Gozes, O., et al., Rapid ai development cycle for the coronavirus (covid-19) pandemic: Initial results for automated detection & patient monitoring using deep learning ct image analysis. arXiv preprint arXiv:2003.05037, 2020.
16. Zhao, J., et al., COVID-CT-Dataset: a CT scan dataset about COVID-19. arXiv preprint arXiv:2003.13865, 2020.
17. Shah, V., et al., Diagnosis of COVID-19 using CT scan images and deep learning techniques. medRxiv, 2020.
18. Wang, S., et al., A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19). MedRxiv, 2020.
19. Xu, X., et al., A deep learning system to screen novel coronavirus disease 2019 pneumonia. Engineering, 2095-8099, 2020.
20. Ronneberger, O., P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. 2015. Springer.
21. Kingma, D.P. and J. Ba, Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.
22. Simonyan, K. and A. Zisserman, Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.
23. Deng, J., et al. Imagenet: A large-scale hierarchical image database. 2009. Ieee.
24. Marcel, S. and Y. Rodriguez. Torchvision the machine-vision package of torch. 2010.