Research Article

Shiva Prasad Koyyada* and Thipendra P. Singh

Ensemble of explainable artificial intelligence predictions through discriminate regions: A model to identify COVID-19 from chest X-ray images

https://doi.org/10.1515/jisys-2023-0163
received September 12, 2023; accepted November 25, 2023

Abstract: In 2019, lung disease severely affected human health and was later renamed coronavirus disease 2019 (COVID-2019). Since then, several research methods have been proposed, such as reverse transcription polymerase chain reaction (RT-PCR), and disease identification through chest X-rays and computed tomography (CT) scans, to help the healthcare sector. RT-PCR was time-consuming when more patients were present, and a CT scan was costly. Several deep-learning (DL) methods were used to identify diseases using computer-aided tools. Among those convolutional neural networks (CNNs), the state of the art was adopted in the machinery to predict cancer. However, there is a lack of explainability (XAI) in how CNN predicts the disease. In this article, we construct XAI ensembles with Local Interpretation Model Agnostic Explanation (LIME), Grad CAM, and a Saliency map. It provides a visual explanation for a DL prognostic model that predicts COVID-19 respiratory infection in patients. Our quantitative experimental results have shown that ensemble XAI with an accuracy of 98.85%, although individual LIME has scored an accuracy of 99.62% on test data, is more reliable since it is the combination of models.

Keywords: XAI, convolutional neural networks, ensembles, lung disease, transfer learning

1 Introduction

Any condition that prevents lung function is called "lung disease." Lung diseases are divided into three categories[1]: airway diseases, lung tissue diseases, and lung circulation disorders. Asthma, pneumothorax, lung cancer, lung infection, and pulmonary edema are prevalent lung illnesses. In all these cases, patients have breathing problems such as shortness of breath, cannot breathe deeply, and have difficulty exhaling. In the same vein, in late 2002, a pneumonia-related disease known as "severe acute respiratory syndrome" (SARS) was reported from Guangdong Province, China, and was officially called SARS in 2003 [1]. In 2019, another severe respiratory disease known as coronavirus disease 2019 (COVID-19) was identified in Wuhan, Hubei Province, China [2]. COVID-19 was initially identified through a reverse transcription polymerase chain

[1] https://medlineplus.gov/ency/article/000066.html.

* **Corresponding author: Shiva Prasad Koyyada**, School of Computer Science, University of Petroleum and Energy Studies, Dehradun, Uttarakhand, 248007, India; Data Sciences, uGDXIT (formerly INSOFE), Hyderabad, 500032, India, e-mail: 500073479@stu.upes.ac.in, shiva.koyyada@gmail.com
Thipendra P. Singh: School of Computer Science, University of Petroleum and Energy Studies, Dehradun, Uttarakhand, 248007, India; School of Computer Science Engineering and Technology, Bennett University, NCR-Delhi, 201310, India, e-mail: thipendra@gmail.com

reaction test, which usually takes at least 24 h to report the results and was time-consuming and limited in kits. Due to this, alternative methods such as chest radiography, and computed tomography (CT) scans were suggested. Chest X-ray (CXR) imaging and CT scans are the standard diagnostic tests for lung diseases. Among these, the CXR method is the cheapest and most accessible to everyone when compared to a CT scan.

CXR images can convey a great deal about a patient's condition; hence, the standard chest radiograph should be reconsidered [3]. Early detection of lung disease is very important to treat patients early and improve the quality of the healthcare system. However, because the availability of radiologists is limited, there is a need to detect lung disease automatically in the absence of experts when immediate treatment is required. Automated medical image analysis began when the first medical image was digitized. It has combined low-level pixel processing (edge and line detector filters, extracting regions) and computational analysis (fitting lines, circles, and ellipses) to develop compound rule-based systems that perform complex operations during the 1970s to the 1990s [4].

Thereafter, designing a computer-aided diagnostic (CAD) system has become essential in supporting medical practitioners in establishing an accurate diagnosis of pneumonia on time [5,6]. With the increasing population and technological advances, it is important to use artificial intelligence (AI) to automate the detection of diseases. Deep convolutional neural networks (DCNNs) became the state-of-the-art tool [7–16] to extract features and detect lung diseases. These help the experts extract complex features, which are very challenging to identify. However, DCNNs are black-box in nature, suffering from the problem of how to explain the model. To mitigate the problem of explainability, another state-of-the-art technology explainable artificial intelligence (XAI) is in use. XAI methods highlight the important patterns in the images. AI has advanced significantly across many industrial areas, particularly since the introduction of deep learning (DL). XAI is the key to unlocking AI and the DL black-box [17]. CXR images have emerged as a helpful tool for the clinical diagnosis and therapy management of COVID-19-related lung illnesses. DL and XAI have gained momentum as DL approaches COVID-19 detection and classification. The goal of this research is to propose and develop an ensemble of XAI techniques for COVID-19 classification models through comparison. Ensembles had great success [18–22] in identifying lung diseases, including COVID-19.

Main contributions of this article can be summarized as follows:

* The proposed method is an ensemble of XAI methods to predict lung diseases through the fusion of global and local features.
* Developing local indicator features through XAI methods such as GradCAM, local interpretation model agnostic explanation (LIME) and Saliency maps.
* A comparison of seven state-of-the-art transfer learning (TL), models was carried out in each XAI method and fusion dataset (FDS).
* Addressing the problem of class imbalance (CI) through a unique multi-stage approach.

This article is designed in the following way: Section 2 describes the review of XAI methods in the field of medical diagnosis, and Section 3 describes the methodology and how ensembles are built. Section 4 describes the experiments and results, and Section 5 is the discussion and way forward.

2 Related work

Researchers have used CXRs and CT scans with DL models for the diagnosis of COVID-19 disease. Many studies are focused on the automation of detecting lung diseases from CXR images since these are less costly. Some of the works developed a framework to identify lung disease and tried to reveal the black-box nature of the DL model. For example, Vidhi [23] has created a system for lung disease identification (COVID-19 vs Normal) from CXR images using layer-wise relevance propagation-based method. They came up with a new metric based on pixel-flipping in order to evaluate the XAI method. The adaptive histogram equalization method was applied during pre-processing on segmented lung images. XAI was used to explain the predictions using different methods. However, no model was built locally on annotated images during the study. Similarly, Naz et al. [24]

has provided an explanation of the classification results of various lung disorders, so that medical practitioners can grasp what causes these ailments. Resnet50 was used to develop a disease classifier, and LIME [25] was used to explain the results. To forecast COVID-19, Gong et al. [26] collected 32 important blood test variables from 1,374 people. During the process, four ensemble learning algorithms were trained, and LIME was used to illustrate the results. In contrast, local features such as SIFT, SURF, BRISK, and KAZE [27] were fed to VGG16 individually to measure the importance while classifying mammographic image.

An ensemble of CNN architectures with three different models of varying kernel sizes (3*3 ,5*5, 7*7) [28] was used to predict pneumonia from CXR images. A sine cosine optimization with DL-based method [29] was developed to identify COVID-19 disease. In similar lines, an ensemble of DL models gave promising results [30]. The EfficientNet model was used as a feature extractor. The Convolutional Block Attention Module and the Wide DenseNet [31] architectures were used to predict tuberculosis (TB). Attention mechanisms are used to allow the network to focus on certain parts of the input data and ignore the others, based on certain criteria. OView-AI system (computer-aided application) [32] used a model trained on Efficient-B7 as the backbone, classifying four diseases: pneumonia, pneumothorax, TB, and lung cancer. Using the dense net as a backbone [33], a network is trained to classify lung disease from a CT scan through a novel loss function. Similarly, ensemble of DL models was applied to classify the lesions in fundus images [34].

Groen et al. [35], using end-to-end DL in their experiments, intend to expand research on the explainability of CAD applications in radiology. From their study, they have identified that 36% of studies have used XAI for visualization. Class activation mapping is the most common technique used. The use of XAI methods has increased [36] in recent years.

To systematically examine the effectiveness of XAI approaches on DL models for pneumonia medical imaging, Zou et al. [37] built a comprehensive XAI evaluation methodology encompassing quantitative and qualitative measures. Ensemble XAI was developed with the supervision of experts and obtained an accuracy of 70%. DL models in AI have been extensively used in a variety of fields, including healthcare and medical imaging. AI must replicate human judgment and interpretation abilities in order to use it as state-of-the-art technology. A thorough examination was conducted by Chaddad et al. [38], and the goal of XAI is to specifically describe the data underlying the black-box model of DL that discloses how judgments are formed. A detailed literature review on the techniques to identify lung diseases is presented by Litjens et al. [4] and Shen et al. [39].

DL with XAI is also used in identifying plant disease [40]. These techniques have achieved great progress in developing a system for detecting plant diseases, and more crucially, GradCAM++ and XAI technique are used to find the disease and emphasize the leaf regions that are most crucial for categorization. Several DL models were applied and tested [41] thoroughly in identifying healthy leaves, including an ensemble of models. Several ensemble models and how XAI methods are used to reveal the black-box nature have been discussed so far with respect to various lung diseases.

Self-attention-based Generator discriminator [42] is designed to extract local features to identify lung cancer using a sunflower optimization algorithm. A multi-stage network was proposed by Vats et al. [43] to detect TB where one stage focuses on localization as well. They also presented a detailed discussion of the activation functions and how these act on data when data are passed through them. Interestingly, Saliency maps were used to segment CXR images [44] and these worked better in reducing the background noise, so they could also be used as a pre-processing step.

The existing research focuses on the entire image to extract the features and classify the disease. In general, a radiologist focuses on the part of the image (say, for example, left/right, lower/upper lobes of CXR image contain blobs) to determine the nature of the disease based on the type of its nature. Very few researchers focused on CXR image localization to predict lung disease. The aforementioned gaps can be addressed with XAI methods. Although some of the works have used XAI for visualization, they have not been used for localization. Inspired by the work of XAI methods, we are proposing a method, which is an integration of LIME, Saliency map, and Grad-CAM methods to identify discriminate regions from CXR images that help to classify lung disease. This will be discussed along with the data and approach in the next section.

3 Dataset and methodology

In this section, we describe the dataset used and the proposed methodology for detecting COVID-19 lung disease. Before delving into the suggested method, we discussed the CNNs and other XAI approaches employed in this method.

3.1 Dataset

Lung disease classification (COVID-19 vs Normal) is achieved by analyzing CXR images. In phase 1, we used 3,500 CXR images (class distribution is shown in Table 1) from the COVID-19 radiography database[2] to train a base classifier, which was adapted from our previous study. Annotated images are generated using the cutting-edge XAI approach LIME in our earlier work [45], where we demonstrated how customized CNN recognized diseases in the region of interest. Furthermore, in this study, we generated annotations for further analysis using the GradCAM and Saliency approaches, and these were archived in the form of numpy arrays for further usage. The dataset will be shared as per the request and based on its use.

Table 1: Number of images used for training and testing in phase 1

S. no.	Data	COVID-19	Normal
1	Traindata	2,396	1,041
2	Testdata	500	300

3.2 CNNs

Instead of computing statistical characteristics from images, a CNN is used to collect feature vectors from images. The first successful CNN [46] was constructed by Lecun for handwritten recognition; however, Fukushima introduced the concept as neocognitron in the 1980s [47]. Convolutional, max pooling, and dense layers make up deep CNNs. Convolution is a two-dimensional weight matrix calculated during training that retains spatial information. Since CNN computes more such weight matrices to extract feature maps from the image, max pooling is used to reduce dimensions and lower the cost of computing weights. Like an artificial neural network, the number of layers in the architecture, and the loss function to optimize the weights, number of epochs, and learning rate are the hyperparameters. A CNN architecture extracting the features from the image is a state-of-the-art technique that has changed the direction of research in image processing; later, it has moved to TL methods such as VGG16 [48], ResidualNet [49], InceptionNet [50], and Densely connected net [51], which are showing remarkable results.

3.3 XAI techniques

Gradient-weighted Class Activation Mapping produces a coarse localization map highlighting the essential regions in the image for predicting the target class using gradients of the target class flowing into the final convolutional layer [52]. The concept originated from the fact that fully connected layers lose spatial information, while convolutional layers retain it. When gradients go into the final convolutional layer, the information

[2] https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database.

is used to assign importance to each neuron. To obtain the localization map, it first computes the gradient of the class score with respect to feature map activations of the convolutional layer and flows back with every layer of convolution. The same process is repeated at each other layer. Gradients are global averages pooled as an aggregation process as they flow back. These techniques are more trustworthy, and they were one of the successful techniques to describe visuals from CNN output.

Saliency map [53] is a ranking-based technique that ranks pixels based on their influence on the class score. Because the class score function is highly nonlinear, identifying the significance of a pixel is difficult. However, the importance of a pixel can be computed using a linear function using the first-order Taylor expansion. $S_c(I) = ||w||^T I + b$, where w is the derivative of S with respect to the image I at the point (image) I_0. $w = \frac{\mathrm{d}S_c}{\mathrm{d}I} \,|_{I_0}$

LIME [25] is another innovative explanation method that learns an interpretable model locally around the predictions in order to explain the predictions of any classifier in an understandable and accurate manner. Interpretable explanations must use a representation that is understandable to humans, regardless of the model’s real attributes. The output is a binary vector showing the presence or absence of a contiguous patch of pixels, via which the model can best predict the class.

There are other XAI methods such as Partial Dependency Plots (PDP), Accumulated Local Effects (ALE), Individual conditional expectation (ICE), SHAP, and ELI5. PDPs [54] illustrate the incremental impact of one or two features on the anticipated outcome of a machine-learning model. ALEs are more similar to PDP with a change in the way of computing the feature importance. It is based on differences in predictions rather than averages [55]. On the other hand, ICE plots [56] are used to assess the impact of a variable on a trained machine-learning model, assuming that all other variables remain constant. The objective of SHAP is to elucidate the prediction of an instance x by calculating the contribution of each feature to the prediction. The SHAP explanation technique [57] calculates the Shapley values using coalitional game theory. The feature values of a data instance function as participants in a coalition. Shapley values provide a method for equitably allocating the “payout” (i.e., the prediction) across the different attributes. It is inspired by a local surrogate model. ELI5 [58] is primarily designed for text data to explain the answers in a simple way.

Among all these methods, LIME, GradCAM, and Saliency mappings are visualization methods that explain images. Methods such as PDP, ALE, and ICE work on lower dimensions and structured data, for example, predicting a customer churn from customer demographical, potential to buy, and job-related features. SHAP has not been applied to any medical images so far. So we employ LIME, GradCAM, and Saliency maps to obtain the region of interest feature that is explained in the next section.

3.4 Proposed method

A collection of XAI models is created and trained to address the problem of binary classification that detects the presence or absence of an individual who has COVID-19. GradCAM, LIME, and Saliency maps are some of the XAI approaches used in this work. To produce ROIs, various XAI approaches are applied, and a CNN model is trained on each output before the final predictions are selected by a majority vote. Decisions made by a group of experts are almost always more trustworthy than decisions made by a single expert. As a result, the predictions are accurate.

The following is the summary of the process: (i) train a CNN model on given input images, (ii) share the stored model and images with XAI method to generate annotations, (iii) fusion of input images and XAI output (generated annotations) and training another custom DCNN on data generated from phase (ii), and (iv) get the predictions from the previous phase for all XAI techniques and finalize the prediction using majority voting. The entire flow is shown in Figure 1.

The process is well explained in phases in Algorithm 1 to train the model, testing is in Algorithm 2, and the schematic is shown in Figure 2. Here, a DCNN with five CNN layers is designed to extract features from CXR images by optimizing the weights with an Adam optimizer and a learning rate of 0.0001. A drop-out layer is introduced to reduce overfitting at the same time.

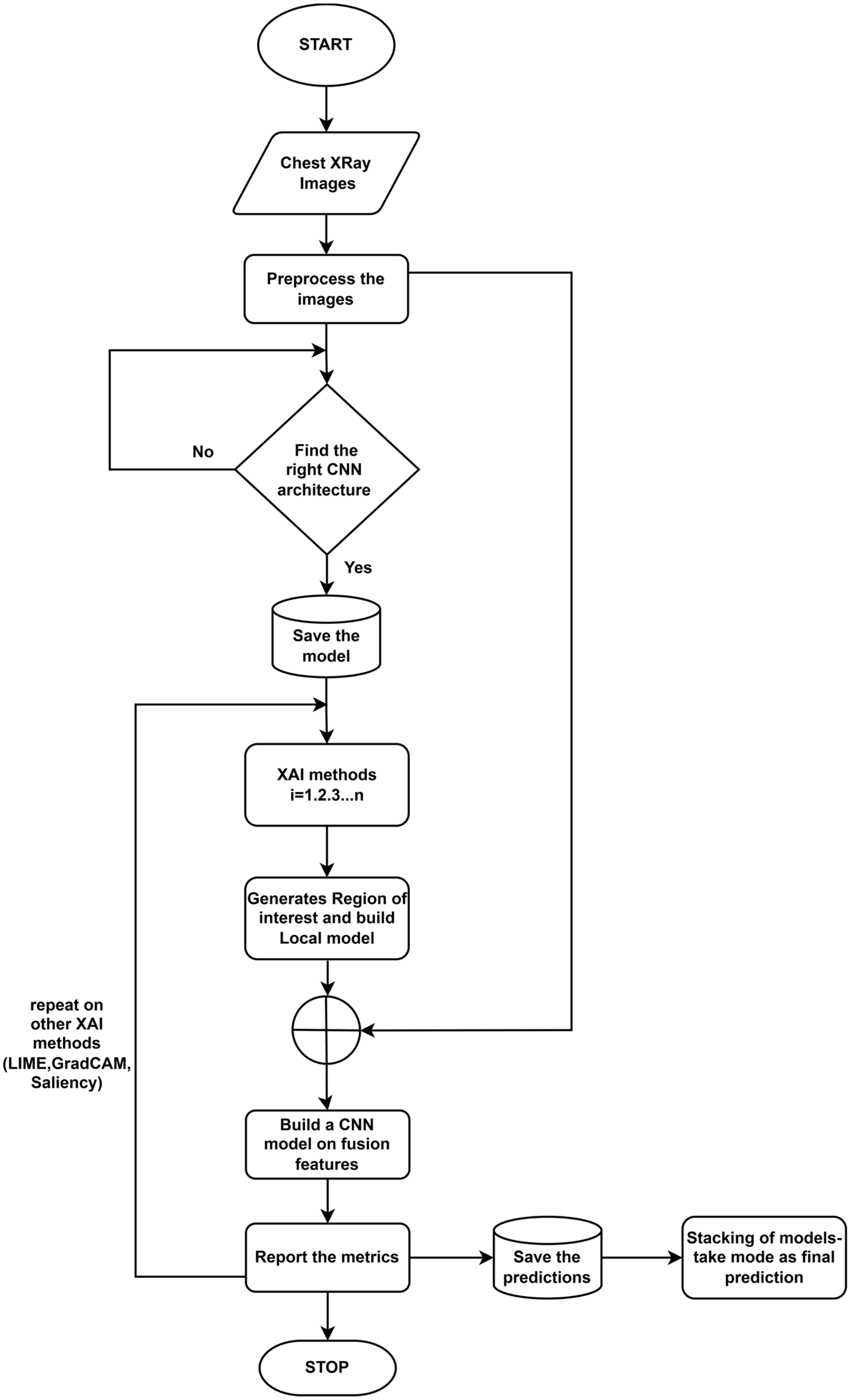

START
Chest XRay Images
Preprocess the images
No
Find the right CNN architecture
Yes
Save the model
XAI methods i=1.2.3...n
Generates Region of interest and build Local model
repeat on other XAI methods (LIME,GradCAM, Saliency)
Build a CNN model on fusion features
Report the metrics
Save the predictions
Stacking of models- take mode as final prediction
STOP

Figure 1: Flowchart that denotes the entire process flow.

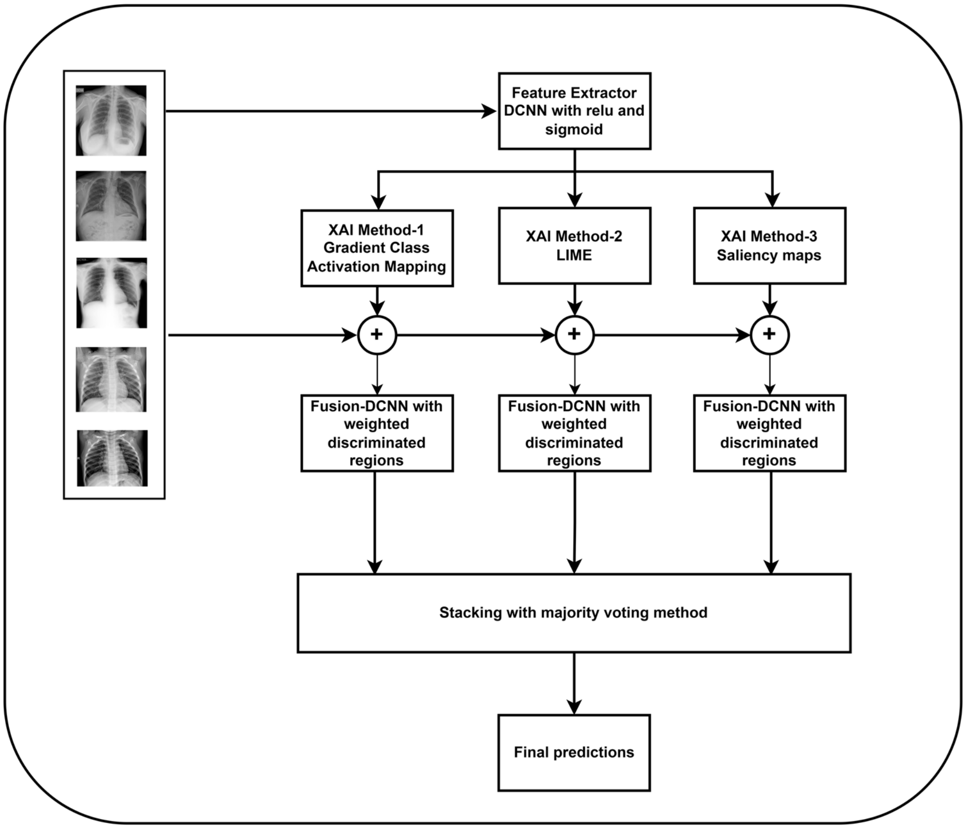

Feature Extractor DCNN with relu and sigmoid
XAI Method-1 Gradient Class Activation Mapping
XAI Method-2 LIME
XAI Method-3 Saliency maps
+
+
+
Fusion-DCNN with weighted discriminated regions
Fusion-DCNN with weighted discriminated regions
Fusion-DCNN with weighted discriminated regions
Stacking with majority voting method
Final predictions

Figure 2: Block diagram for proposed method: Ensemble XAI; DCNN: Deep convolutional neural net; F-DCNN: Fusion DCNN.

Algorithm 1: Ensemble of XAI models: Training

```
Read CXR images of 224*224 resolution.
Load customized trained model(base model) from h5 object.
    Phase 1 - Generating ROIs-Local features from CXR images
while All images exhausted do
    Input model and image to the XAI model to generate ROI mask.
    $i \leftarrow XAImethod$
    if $i = GradCAM$ or $i = Saliencymap$ then
        Send the image to the XAI method to get the ROI mask
        Apply Otsu’s method on ROI to get the right segments
    else
        Send the image to the XAI method to get the ROI mask
    end if
    Write the roi generated image to disk.(named as local dataset (LD)
end while
    Phase 2 - Training CNN model on local features- Local models
while On each dataset produced in the previous step do
    Train a CNN model.
    Store the model as an h5 object.
end while
    Phase 3 - Fusion of original images with extracted local regions
while On each LD(s) do
```

```
        read images from LD and original dataset
        while All images exhausted do
            Point wise pixel addition local feature and original image
            prepare a numpy array
        end while
        Store numpy array for each - FDs
    end while
      Phase 4 - Training a CNN model on combined dataset
    while On each FD(s) do
        Train a CNN model
        Store it as h5 object (Fusion model-FM)
        Get the predictions
        Take the mode of predictions as the final output
    end while
```

DCNN feature extractor is a binary classifier with a fully connected layer plus a sigmoid activation function that outputs the probability. The trained model, along with images given as input to the XAI method, produces the described region. These discriminated regions can be given more weight while combining image features with Fusion DCNN, which has a similar architecture. Since various models produce different probabilities, stacking is performed on the output with a mode of voting.

Algorithm 2: Ensemble of XAI models Testing

```
Read test CXR images of 224*224 resolution.
Load customized trained model(base model) from h5 object.
Load three local models from the h5 object.
while All images exhausted do
    while each of three XAI method(s) do
        produces Local features using the XAI model.
        Fusion with original data
        Apply fusion model to get the prediction.
    end while
    Take the mode of all as the final prediction
end while
```

The time complexity for the proposed approach is estimated in a model-specific way [59] as per the architecture in a total of three parts. One is training the base model before applying XAI methods. Second is generating rois from model and XAI methods. Third is training on FDs. The first and third used the same architecture with the goal of capturing the features whatever is not captured before fusion. It can be defined [60] in the following way:

$$\sum_{i=1}^{d} f(m_i, n_i, f_i, s_i), \tag{1}$$

where "m" and "n" denote the size of features, "f" is the number of kernels, "s" is the size of the kernel, and "d" denotes the depth of the network. The second part again consists of three parts since the XAI methods are different. For simplification k_1, k_2, and k_3 denote the time taken by three XAI methods, respectively, to produce a region of interest. "k" is defined as:

$$k = f(XAI \text{ method, Model}), \tag{2}$$

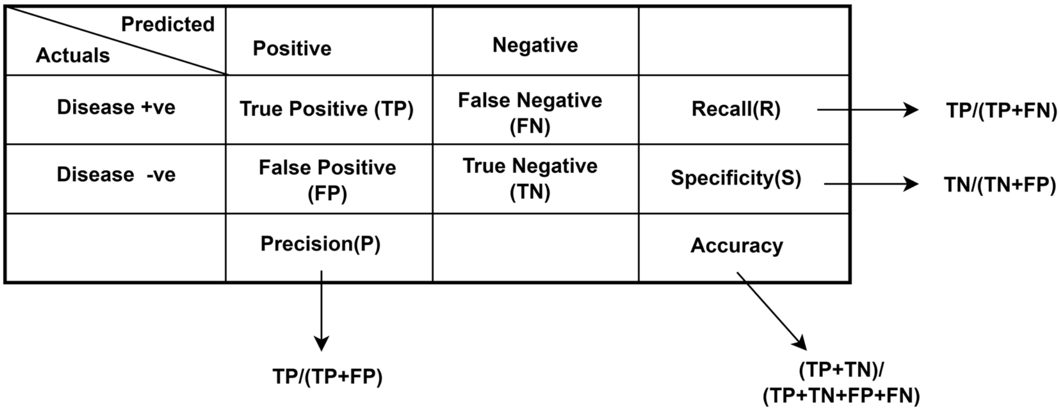
and then, the summation is defined as follows:

$$\sum_{i=1}^{x} f(k_i). \tag{3}$$

In the aforementioned notation (2), the “Model” is one of the components, because the image and model will be passed to the XAI method. XAI method time depends on the architecture of the model as well. “x” is the number of XAI methods: in this case, it is three. The total time is the sum of (1) and (3). We have also measured the time for an epoch in each of these XAI methods. It excludes read/write intermediate results.

4 Experiments and results

The problem is formulated as a binary classification problem since it is classifying lung diseases COVID-19 vs Normal. Combined with original data, XAI-generated data is the input for the models, which consists of 224*224 resolution images. Any classification problem is always measured with a confusion matrix as shown in Figure 3 metrics such as accuracy, recall, precision, and *F*1 score. Accuracy is defined as the number of correct predictions over all samples. The recall is a true-positive ratio, which is nothing but how many correct positives out of all actual positives. Precision is the predicted positive ratio which is nothing but how many true positives out of all predicted positives. Specificity is a true negative ratio, which is defined as the number of true-negatives out of all actual negative images. The *F*1 score is the geometric mean of precision and recall. Along with accuracy, either recall or specificity will be used in the evaluation, depending on the importance given to the positive class or negative class. The recall will be given priority to penalize false negatives, whereas specificity to penalize false positives.



Predicted Actuals	Positive	Negative	
Disease +ve	True Positive (TP)	False Negative (FN)	Recall(R)
Disease -ve	False Positive (FP)	True Negative (TN)	Specificity(S)
	Precision(P)		Accuracy

TP/(TP+FN)

TN/(TN+FP)

TP/(TP+FP)

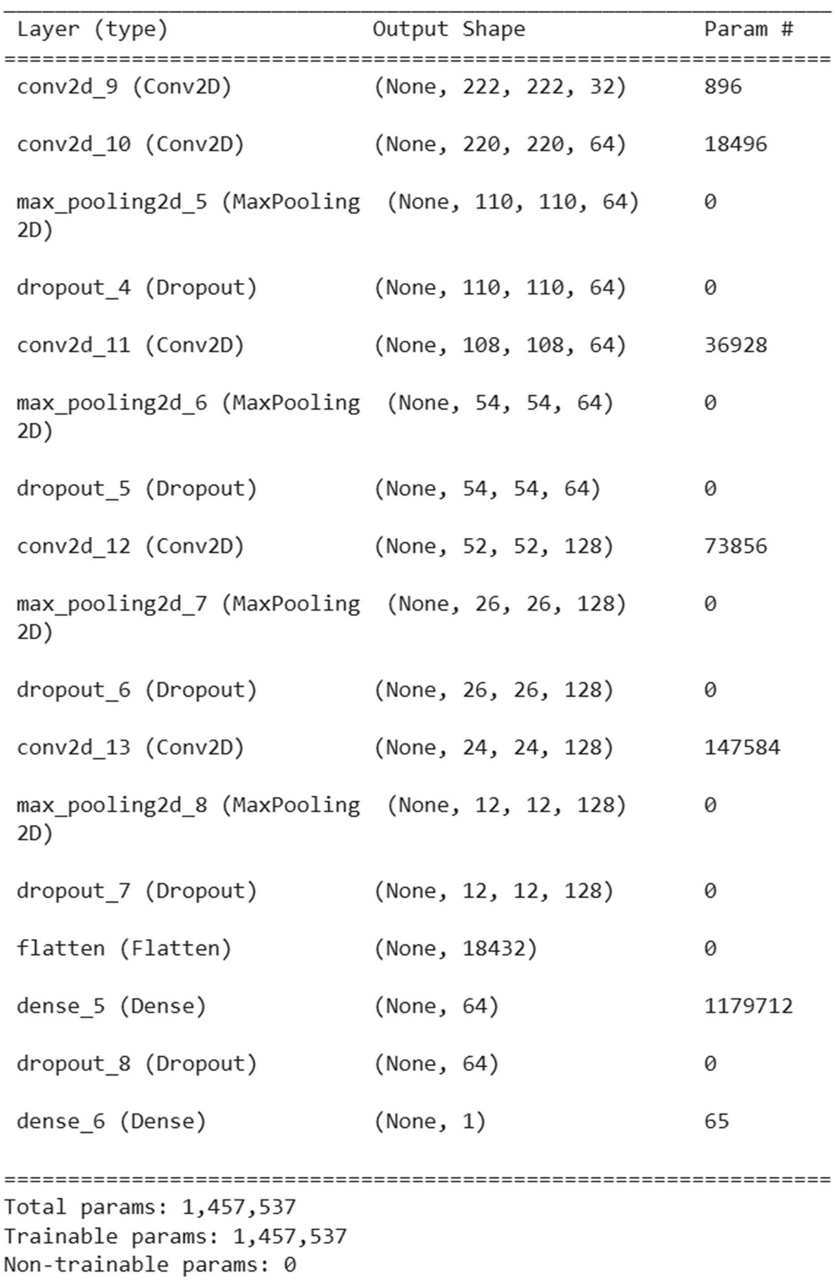
(TP+TN)/
(TP+TN+FP+FN)

Figure 3: Confusion matrix.

4.1 Ensemble of XAI methods

In the proposed method, COVID-19 and Normal CXR images of 224*224 size have been taken as input. Experiments were conducted on google collab[3]. In all the experiments, DCNN architecture as shown in Figure 4 was used, where the input layer has a dimension of 224*224.

[3] https://colab.reasearch.google.com.



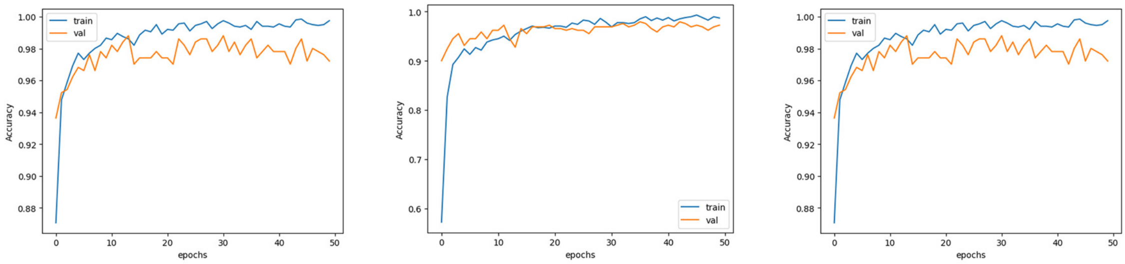
```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_9 (Conv2D)           (None, 222, 222, 32)      896

 conv2d_10 (Conv2D)          (None, 220, 220, 64)      18496

 max_pooling2d_5 (MaxPooling  (None, 110, 110, 64)     0
 2D)

 dropout_4 (Dropout)         (None, 110, 110, 64)      0

 conv2d_11 (Conv2D)          (None, 108, 108, 64)      36928

 max_pooling2d_6 (MaxPooling  (None, 54, 54, 64)       0
 2D)

 dropout_5 (Dropout)         (None, 54, 54, 64)        0

 conv2d_12 (Conv2D)          (None, 52, 52, 128)       73856

 max_pooling2d_7 (MaxPooling  (None, 26, 26, 128)      0
 2D)

 dropout_6 (Dropout)         (None, 26, 26, 128)       0

 conv2d_13 (Conv2D)          (None, 24, 24, 128)       147584

 max_pooling2d_8 (MaxPooling  (None, 12, 12, 128)      0
 2D)

 dropout_7 (Dropout)         (None, 12, 12, 128)       0

 flatten (Flatten)           (None, 18432)             0

 dense_5 (Dense)             (None, 64)                1179712

 dropout_8 (Dropout)         (None, 64)                0

 dense_6 (Dense)             (None, 1)                 65

=================================================================
Total params: 1,457,537
Trainable params: 1,457,537
Non-trainable params: 0
```

Figure 4: Layer-wise detail of the DCNN model.

Once the DCNN model is trained on COVID-19 and Normal classes, images and model were fed to the XAI technique, for example, GradCAM, through which annotated images were generated. The annotated images generated using GradCAM and Saliency maps are preprocessed further with Otsu's thresholding method to get the segments. Otsu is a segmentation algorithm that generates lower and upper thresholds to segment the image based on the continuity of pixels.[4] Similarly, LIME is another XAI model applied to this problem [45]. When we use LIME, the feature weights will vary between 0.0001 and 0.00000001. Fusion is a simple process of the addition of pixels, which intuitively gives more weight to the important regions though it does not show much interpretation when we plot the fusion image. Of course, it can be done through normalization but it

[4] https://learnopencv.com/otsu-thresholding-with-opencv/.

does not convey anything to the human eye. Three XAI techniques (GradCAM, LIME, and Saliency maps) were used during experimentation to generate critical regions. Each generates local discriminate regions that are nothing but masked images and were used further to train a model along with original data. Error curves for these are shown in Figure 5.



(a) (b) (c)

Figure 5: Error curves: a–c are generated while training on the GradCAM, Saliency, and LIME FDSs.

The results reported in Table 2 for an ensemble of XAI are predicting all positive cases correctly without missing anything although the proposed method is on par with individual models in other metrics. The number of XAI methods in ensembles can be 3, 5, 7, etc. any odd number that avoids a tie between the models.

Table 2: Metrics captured from the models built on XAI generated annotations when combined with original data; in train column, one indicates the train data and zero indicates the test data, Ensemble are preferred over the other models, metrics such as Accuracy, Recall and Precision are indicated in bold

S. no.	XAI method	Train	Accuracy	Recall	Precision	*F*1 score
		1	99.44	99.45	99.8	1.00
1	GradCAM	0	98.79	99.23	99.1	0.99
		1	99.24	99.47	99.1	0.99
2	Saliency	0	98.05	98.22	98.2	0.98
		1	99.96	100.00	100.0	1.00
3	LIME	0	99.62	100.00	99.4	1.00
4	Ensembles	0	**98.85**	**97.88**	**100.0**	0.99

The experiments were conducted on Google Colab with default setting (12.7 GB RAM,107.7GB disk space), and it took an average of 12.56, 7.59, and 12.68 s for GradCAM, Saliency, and LIME per each epoch, respectively, whereas the base model took 3.86 s. The average time period here is for the number of times the experiment (10 runs) was conducted.

4.2 Correlation analysis with TL techniques

TL [61] has been applied to many real-world problems to address the following: (i) insufficient labeled data, (ii) incompatible computational power, and (iii) distribution mismatch. TL methods are DCNNs trained on a huge database for a classification problem. Since it is deep, the network can be used to extract features from any given image instead of images from the same distribution. This has hinged on the researchers training another network by taking the output feature vectors from DCNN, which we say it as TL method. Here, we are using the weights of the pre-trained network to generate the feature vectors. Various pre-trained networks such as

VGG16 [48], ResNet [49], MobileNet [62], Xception [63], EfficientNet [64], DenseNet [51], Inception [50], and ConvNext [65] are available with different architectures and can be reused with imagenet pre-trained weights. These architectures had a difference in their design of architecture. For example, ResNet uses residual blocks, XceptionNet uses depth-wise separable convolutions, and DenseNet has dense blocks. A common concept among these is that they are built on top of convolutional layers but with different filter sizes. The recent ConvNext architecture is trained on Resnet based on the vision transformers concept. In this work, the authors have used some of the TL methods by restricting themselves to one version from every family of TL methods. The experiments were conducted on the FDs. A TL method is trained on FDs with binary cross-entropy loss, Adam optimizer for 50 epochs on every XAI-generated output.

TL methods have also been applied to the original dataset and recorded as base results to compare with the results in Table 3. The metrics are reported in Table 4. One can compare these results with the average metric computed across all the TL methods built on FDs, which is slightly higher than the individual models on the original data. It will be efficient if we compare it at the individual model level. However, the panel of the TL model's decisions is much more reliable than the base one. This analysis can be done by comparing with the results from Table 2 and claim that the ensemble of customized model results is consistent.

Some of the assumptions were made while training all the networks for standardization purposes. Every network is trained for 50 epochs only with a learning rate of 0.001, an optimizer of Adam, and a patience of two after several trials. No other parameter tuning was done to improve the results or avoid overfitting. With the approach adopted, we can claim that some of the networks are weak classifiers that may tend to do well on a few data points when compared to others, which is actually the concept of ensembles. Some of the error curves are shown in Figure 6.

4.3 Handling CI

One of the major problems in machine learning is CI. CI is defined as there will be fewer samples (<10%) of the total when compared to the opposed class. There are a number of methods available to address CI on structured data such as under-sampling, over-sampling, and generation of additional data. However, there is no method to deal with images. A method that we have implemented on structure data [66] experimented on images is able to obtain significant results. The detailed architecture is shown in Figure 7.

The images are sampled to create CI data with 100 images from COVID-19 and 900 from Normal class with the intention of creating a 90:10 ratio. Here, the minority class samples are from the COVID-19 class; in general, positive samples are rare for any disease. The experiments are conducted in the following way: (i) trained a customized model on the sample dataset with the parameters stated earlier, (ii) created mutual disjoint datasets (MDS) with repeated sampling from the majority class while combining with minority class samples, (iii) trained individual models on each MDS and saved the predictions, (iv) consolidated the predictions from all MDS with two-stage voting for minority class samples, and (v) comparison of the results before and after applying the method. The experiments have a choice of choosing models between customized models and TL methods. However, TL method is a good choice as there are fewer images. The results were recorded during the training original dataset, and MDS is given in Table 5.

This dataset has less than 10% of samples in the COVID-19 class. When creating MDS, it has been brought to 40% of the total with the undersampling method. The experiments were conducted on the original dataset before creating MDS and can be compared with the results generated from MDS. We can observe that all methods gave results on par with the based metric. Among TL methods, VGG16 has received low scores. However, it has performed well on test data using the proposed method.

Table 3: Metrics recorded after applying various TL methods on different XAI-generated FDs: in train column, one indicates the train data and zero indicates the test data, Different models (TL method) are performing well on test data when different XAI methods are used with respect to Accuracy, Precision and F1 score and these are mentioned in bold

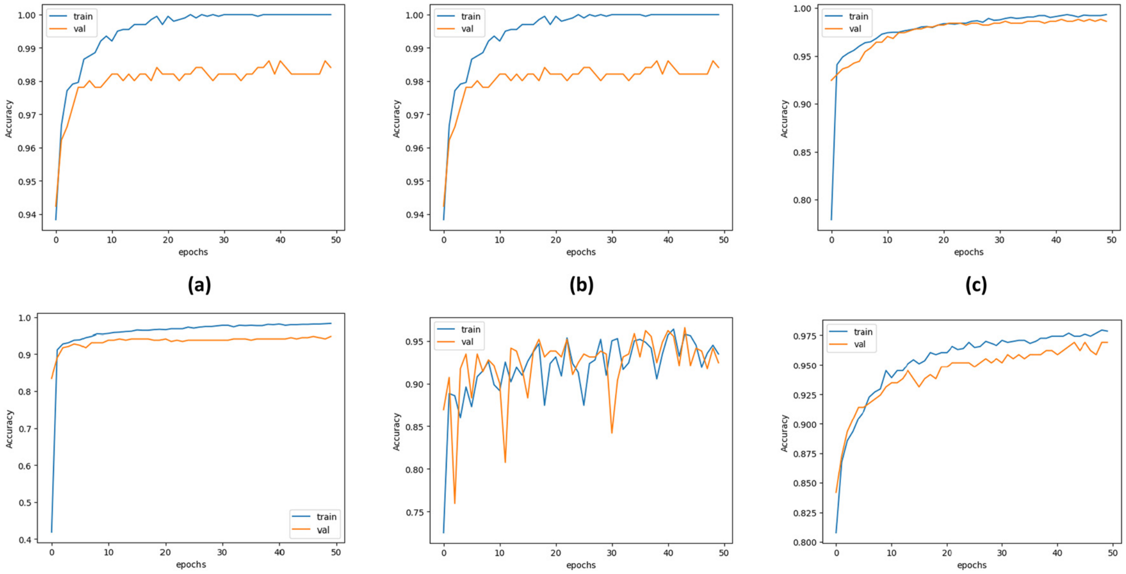
S. no.	XAI method	TL	Train	Accuracy	Recall	Precision	*F*1 score
1	GradCAM	Resnet50v2	1	99.52	99.72	99.61	1.00
2	GradCAM	Resnet50v2	0	97.74	98.23	98.67	0.98
3	GradCAM	MobileNetv3small	1	99.20	99.56	99.34	0.99
4	GradCAM	MobileNetv3small	0	**99.44**	99.45	**99.78**	**1.00**
5	GradCAM	Xception	1	99.48	99.56	99.72	1.00
6	GradCAM	Xception	0	96.37	96.49	98.54	0.98
7	GradCAM	ConvNextTiny	1	99.68	99.73	99.84	1.00
8	GradCAM	ConvNextTiny	0	97.66	98.73	97.95	0.98
9	GradCAM	EfficientNetV2S	1	99.68	99.78	99.78	1.00
10	GradCAM	EfficientNetV2S	0	**99.19**	99.33	**99.55**	**0.99**
11	GradCAM	DeseNet121	1	99.08	98.96	99.78	0.99
12	GradCAM	DeseNet121	0	**99.68**	99.77	**99.77**	**1.00**
13	GradCAM	VGG16	1	99.68	99.78	99.78	1.00
14	GradCAM	VGG16	0	98.87	99.31	99.08	0.99
15	LIME	Resnet50V2	1	99.76	99.89	99.78	1.00
16	LIME	Resnet50V2	0	**99.60**	99.67	**99.78**	**1.00**
17	LIME	MobileNetv3small	1	84.10	96.47	84.13	0.90
18	LIME	MobileNetv3small	0	82.10	96.22	81.65	0.88
19	LIME	Xception	1	99.96	99.95	100.00	1.00
20	LIME	Xception	0	**99.84**	99.77	**100.00**	**1.00**
21	LIME	ConvNextTiny	1	94.46	94.82	97.48	0.96
22	LIME	ConvNextTiny	0	94.92	95.63	97.26	0.96
23	LIME	EfficientNetV2S	1	96.66	97.03	98.33	0.98
24	LIME	EfficientNetV2S	0	95.73	97.21	96.89	0.97
25	LIME	DeseNet121	1	99.88	99.89	99.94	1.00
26	LIME	DeseNet121	0	**99.92**	100.00	**99.89**	**1.00**
27	LIME	VGG16	1	98.77	99.28	99.01	0.99
28	LIME	VGG16	0	**99.59**	99.89	**99.55**	**1.00**
29	Saliency	Resnet50V2	1	92.35	99.34	87.94	0.93
30	Saliency	Resnet50V2	0	91.77	99.23	87.39	0.93
31	Saliency	MobileNetv3small	1	97.66	97.44	98.19	0.98
32	Saliency	MobileNetv3small	0	96.93	96.79	97.31	0.97
33	Saliency	Xception	1	98.28	97.86	98.78	0.98
34	Saliency	Xception	0	95.40	93.63	98.20	0.96
35	Saliency	ConvNextTiny	1	98.42	97.53	99.47	0.98
36	Saliency	ConvNextTiny	0	99.16	98.96	99.48	0.99
37	Saliency	EfficientNetV2S	1	93.04	96.80	93.90	0.95
38	Saliency	EfficientNetV2S	0	93.23	97.47	93.19	0.95
39	Saliency	DeseNet121	1	95.81	92.15	99.86	0.96
40	Saliency	DeseNet121	0	95.96	92.56	100.00	0.96
41	Saliency	VGG16	1	98.77	99.28	99.01	0.99
42	Saliency	VGG16	0	**99.59**	99.89	**99.55**	**1.00**

5 Discussion and analysis

The annotations produced by XAI methods such as GradCAM, LIME, and Saliency maps are subject to the base model built on the data, so it is critical to produce trustworthy interpretations based on a fixed model. Especially in contrast to individual models, ensemble XAI has the benefit of stable interpretation. It is because each model might be focusing on different roi while providing the final outcome. Additionally, due to the presence of text, catheters, or lines in the X-ray image, the base heat maps produced by GradCAM and LIME rarely highlight the regions outside the lungs. Even though this distinct and interrupting area may be a sign of

Table 4: Metrics generated from TL methods- on original images of 224*224 resolution: in train column, one indicates the train data and zero indicates the test data

S. no.	TL method	Train	Accuracy	Recall	Precision	*F*1 score
1	Resnet50v2	1	99.96	100.00	99.94	1.00
2	Resnet50v2	0	99.68	99.67	99.89	1.00
3	MobileNetv3small	1	86.88	97.48	86.25	0.92
4	MobileNetv3small	0	87.02	96.84	86.59	0.91
5	Xception	1	99.92	99.89	100.00	1.00
6	Xception	0	99.44	99.55	99.66	1.00
7	ConvNextTiny	1	93.04	96.80	93.90	0.95
8	ConvNextTiny	0	93.23	97.47	93.19	0.95
9	EfficientNetV2S	1	98.21	98.83	98.67	0.99
10	EfficientNetV2S	0	98.15	99.01	98.47	0.99
11	DenseNet121	1	99.92	99.95	99.95	1.00
12	DenseNet121	0	99.84	99.78	100.00	1.00
13	VGG16	1	99.01	99.12	99.50	0.99
14	VGG16	0	98.47	98.55	99.32	0.99



(a) (b) (c)

(d) (e) (f)

Figure 6: Error curves: a–c are generated while training on the GradCAM FD; d–f are generated while training on the Saliency FD. (a) GradCAM-VGG16, (b) GradCAM-Resnet, (c) GradCAM-MobileNet, (d) Saliency-VGG16, (e) Saliency-ResNet, and (f) Saliency-Mobilenet.

a serious lung condition, it is not useful for making decisions. XAI models have produced an average accuracy of 99.55%, 98.83% recall of 99.64%, 99.15% precision of 99.60%, 98.91% and *F*1 score of 1, 0.99 on train and test data, respectively. While we are addressing an ensemble of XAI methods, our experimentation went on to TL methods as these are state-of-the-art techniques. ResNet50v2, MobileNetv3small, Xception, ConvNextTiny, EfficientNetV2S, DenseNet121, and VGG16 are the seven different techniques. The criteria behind the selection of these models, each from a different family of models, are different at their architectural level. Each TL method performed differently for each FD, for example, MobileNetv3small has a score of 99.44% for GradCAM, and ResNet50v2 has produced a better result of 99.60% in the case of LIME as VGG16 with 99.59% accuracy for Saliency maps. The topic of whether it is wise to rely on the TL approaches or the customized method has been raised, and the solution may lie in a combination of the two. It requires a great deal of experimentation and

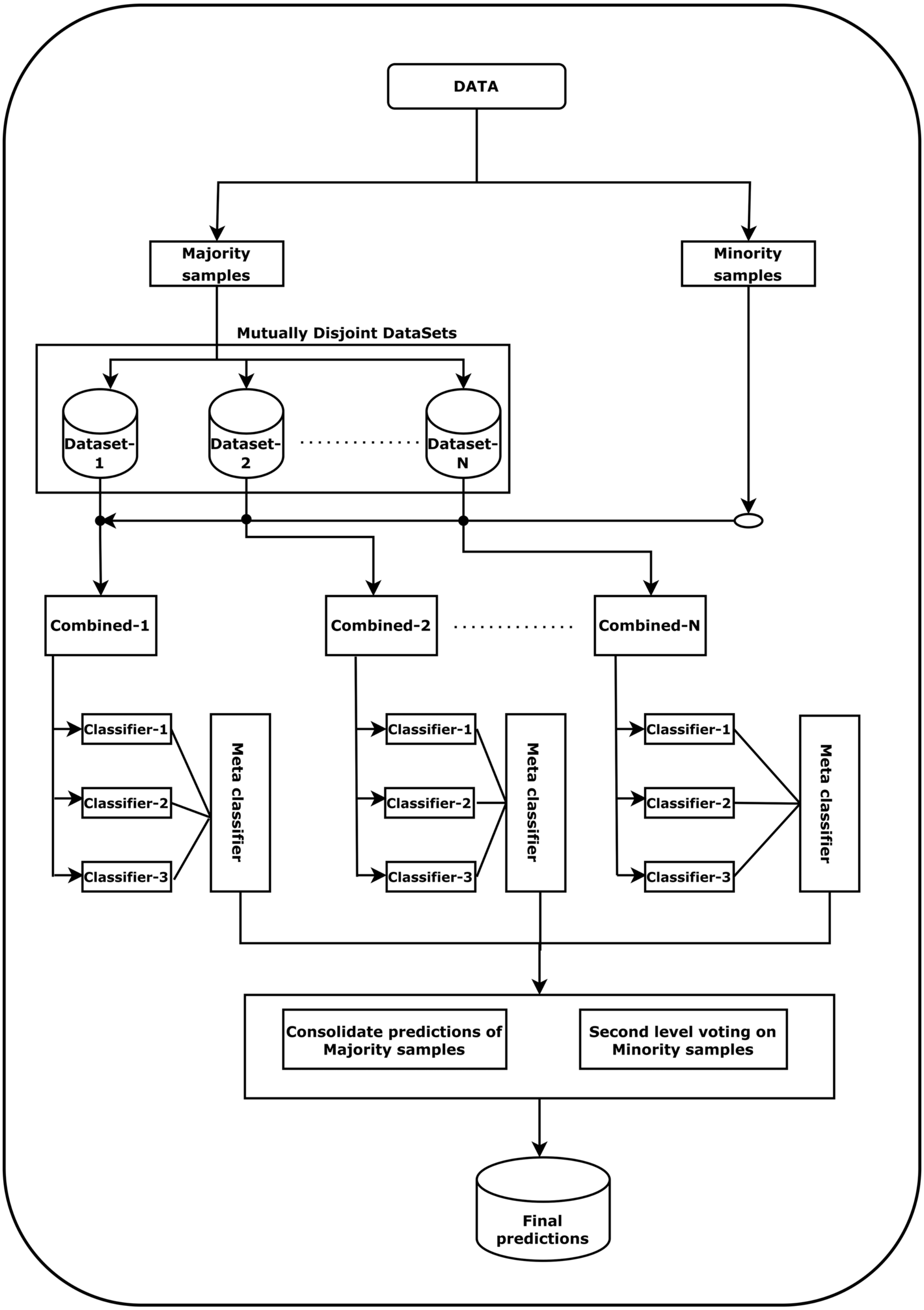


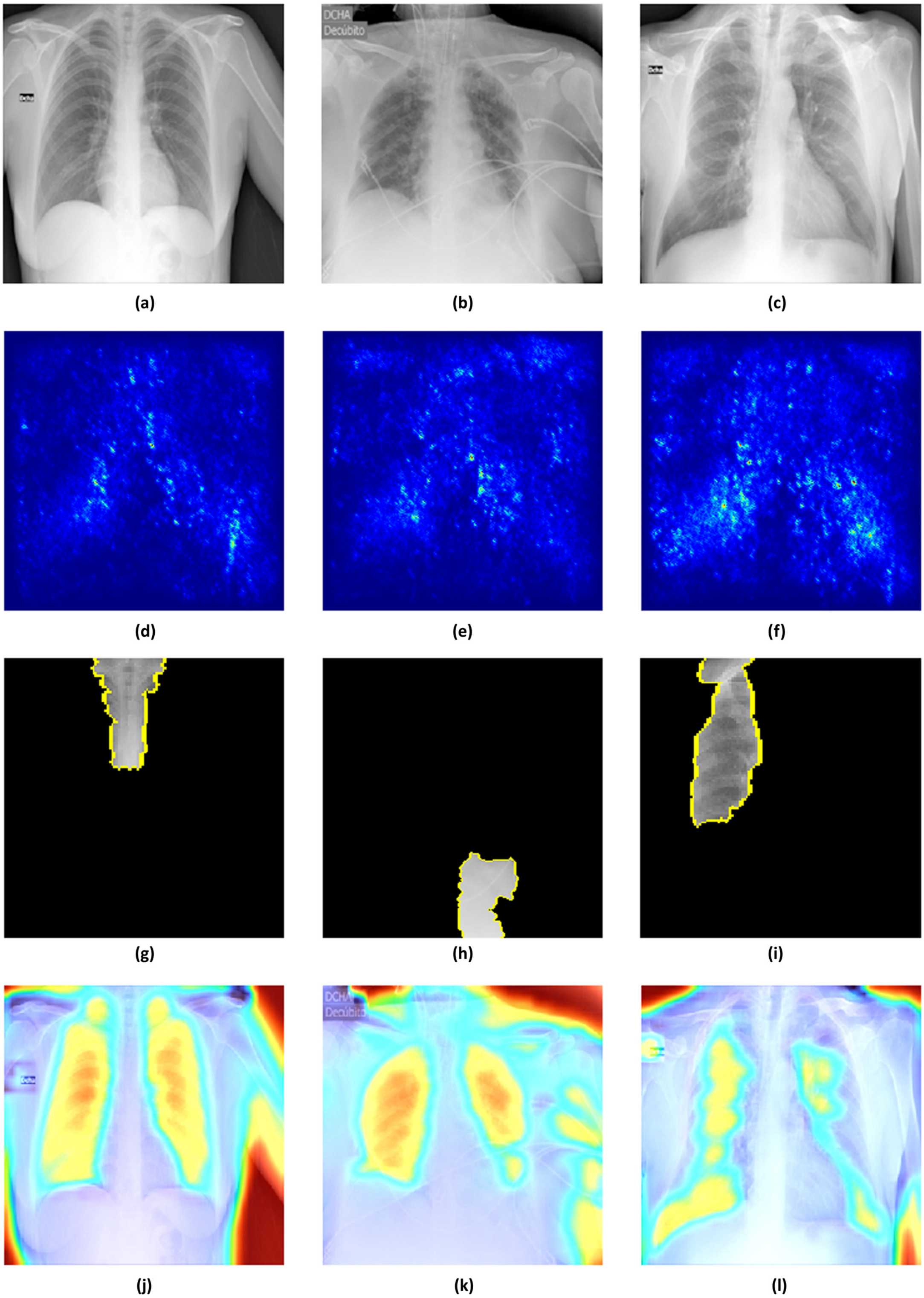
Figure 7: A multi-stage approach to handle CI on images [66].

Table 5: Metrics generated from TL methods- on original images of 224*224 resolution: in train column, one indicates the train data and zero indicates the test data and metrics recorded on test data with proposed method are mentioned in the bold

S. no.	Model	Data	Train	Accuracy	Recall	Precision	*F*1
1	Customized	Original data (90:10)	1	98.17	99.88	98.10	0.99
2	Customized	Original data (90:10)	0	99.04	98.91	100.00	0.99
3	Customized	MDS1	1	97.86	94.44	100.00	0.97
4	Customized	MDS1	0	96.15	90.00	100.00	0.95
5	Customized	MDS2	1	99.15	100.00	98.59	0.99
6	Customized	MDS2	0	96.15	100.00	94.44	0.97
7	Customized	MDS3	1	99.15	100.00	98.62	0.99
8	Customized	MDS3	0	96.15	100.00	93.33	0.97
9	Customized	Testdata-proposed method	0	**97.61**	**98.13**	**98.50**	**0.98**
10	Customized	Original test data	0	98.14	98.88	98.51	0.99
11	DenseNet	Original data (90:10)	1	99.68	99.84	99.84	1.00
12	DenseNet	Original data (90:10)	0	99.71	99.67	100.00	1.00
13	DenseNet	MDS1	1	99.43	98.99	100.00	0.99
14	DenseNet	MDS1	0	100.00	100.00	100.00	1.00
15	DenseNet	MDS2	1	97.70	96.36	100.00	0.98
16	DenseNet	MDS2	0	97.67	96.43	100.00	0.98
17	DenseNet	MDS3	1	99.43	99.12	100.00	1.00
18	DenseNet	MDS3	0	95.35	93.88	97.87	0.96
19	DenseNet	Testdata-proposed method	0	**96.28**	**89.83**	**98.15**	**0.94**
20	DenseNet	Original test data	0	96.54	89.26	100.00	0.94
21	Resnet	Original data (90:10)	1	100.00	100.00	100.00	1.00
22	Resnet	Original data (90:10)	0	99.71	99.67	100.00	1.00
23	Resnet	MDS1	1	100.00	100.00	100.00	1.00
24	Resnet	MDS1	0	100.00	100.00	100.00	1.00
25	Resnet	MDS2	1	99.42	99.07	100.00	1.00
26	Resnet	MDS2	0	100.00	100.00	100.00	1.00
27	Resnet	MDS3	0	99.43	99.05	100.00	1.00
28	Resnet	MDS3	0	97.67	96.55	100.00	0.98
29	Resnet	Testdata-proposed method	0	**97.91**	**93.86**	**99.07**	**0.96**
30	Resnet	Original test data	0	94.72	85.71	100.00	0.92
31	VGG16	Original data (90:10)	1	94.80	94.62	100.00	0.97
32	VGG16	Original data (90:10)	0	92.96	92.59	100.00	0.96
33	VGG16	MDS1	1	85.63	81.62	100.00	0.90
34	VGG16	MDS1	0	83.72	77.78	100.00	0.88
35	VGG16	MDS2	1	87.93	83.33	100.00	0.91
36	VGG16	MDS2	0	86.05	82.09	100.00	0.90
37	VGG16	MDS3	1	84.48	80.58	100.00	0.89
38	VGG16	MDS3	0	76.74	70.59	100.00	0.83
39	VGG16	Testdata-proposed method	0	**73.94**	**52.43**	**100.00**	**0.69**
40	VGG16	Original test data	0	59.31	41.38	100.00	0.59

takes quite a while to make predictions based on actual facts. A lot of experiments were conducted to handle CI and some of them showed better results; however, it is to be seen in depth.

There are limitations of the study, and experiments were conducted on around 3,500 images. It raises the question of generalizing the solution. However, it provides base metrics for future studies. Interpretation of X-ray images for nonmedical experts is a tedious task, while expert radiologists can interpret the XAI output from X-rays. Manual inspection of each image is a tedious and time-consuming task. This showcases the need for XAI methods to generate annotations. A major roadblock for an individual researcher for experimentation is computing power. The proposed method could be extended by including more and more images having computing power to overcome the impediment. Some of the XAI-generated images are shown in Figure 8 for the respective methods (Saliency, LIME, and Grad-CAM).



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

Figure 8: XAI-generated images: a–c are original images; d–f are Saliency maps; g–i are the LIME-generated masks; and j–l are the GradCAM output: (a) COVID-1745, (b) COVID-2635, (c) COVID-2865, (d) COVID-1745, (e) COVID-2635, (f) COVID-2865, (g) COVID-1745, (h) COVID-2635, (i) COVID-2865, (j) COVID-1745, (k) COVID-2635, and (l) COVID-2865.

6 Conclusion and the way future work

The authors have proposed a versatile approach to computing local features through XAI methods and shown better accuracy of 98.85% and 99.62% for ensemble XAI and LIME, respectively. With the TL approaches, XAI-Xception net gave the best test result of 99.84% and while handling CI custom network gave 97.61% on test data. As the number of options is increasing as technology progresses, one can opt for heterogeneous approaches to build systems to predict lung disease in the near future. As the research grows, these XAI method-generated images to be inspected by radiologists and come up with the right annotations may increase the trust in the patients and healthcare community to make use of the automation process. The proposed approach may be extended to other disease images and in other modalities such as CT scans and magnetic resonance imaging (MRI). There is a possibility of building a segmentation model on each XAI method-generated output. Further ensemble of segmentation may provide the right annotations instead of depending on XAI methods once there is a well-trained segmentation model similar to the study by Ronneberger et al. [67]. Unet architecture is a network and a training strategy that has a contracting path to capture context and a symmetric expanding path that enables precise localization. Training such a network requires image masks along with images that need to be verified by experts. Here, an attempt is made to build Ensemble XAI methods to predict lung disease (COVID-19 vs Normal), and it has produced sustainable results.

Funding information: No funding was received to assist with the preparation of this manuscript.

Author contributions: All authors have accepted responsibility for the entire content of this manuscript and approved its submission. Shiva prasad Koyyada performed the experiments and drafted the manuscript. Thipendra P Singh has supervised the entire process and reviewed it.

Conflict of interest: The authors have no competing interests to declare that are relevant to the content of this article.

Ethical statement: No ethical approval was required for this work because the dataset used to build the xai image is freely available online with proper citation, and no explanations were provided for individual images.

Data availability statement: The images generated through the XAI method in this research will be available at a reasonable request by mentioning the reason for academic research.

References

[1] Thomas G, Ksiazek DE. A novel coronavirus associated with severe acute respiratory syndrome. New England J Med. 2003;348:1953–66. http://www.nejm.org.

[2] Wu F, Zhao S, Yu B, Chen YMM, Wang W, Song ZGG, et al. A new coronavirus associated with human respiratory disease in China. Nature. 2020;579:265–9.

[3] Van Ginneken B, Ter Haar Romeny BM, Viergever MA. Computer-aided diagnosis in chest radiography: a survey. IEEE Trans Med Imag. 2001;20(12):1228–41.

[4] Litjens G, Kooi T, Bejnordi BE, Setio AAA, Ciompi F, Ghafoorian M, et al. A survey on deep learning in medical image analysis. Med Image Anal. 2017 July;42:60–88.

[5] Rajaraman S, Guo P, Xue Z, Antani S. A deep modality-specific ensemble for improving pneumonia detection in chest X-rays. Diagnostics. Switzerland: Basel; 2022 June. p. 12.

[6] Ravi V, Narasimhan H, Pham TD. A cost-sensitive deep learning-based meta-classifier for pediatric pneumonia classification using chest X-rays. Expert systems. 2022 Aug;39:e12966.

[7] Ozturk T, Talo M, Yildirim EA, Baloglu UB, Yildirim O, Acharya UR. Automated detection of COVID-19 cases using deep neural networks with X-ray images. Comput Biol Med. 2020;121:103792.

[8] Apostolopoulos ID, Mpesiana TA. Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks. Phys Eng Sci Med. 2020 June;43:635–40.

[9] Chowdhury MEH, Rahman T, Khandakar A, Mazhar R, Kadir MA, Mahbub ZB, et al. Can AI help in screening viral and COVID-19 Pneumonia? IEEE Access. 2020;8:132665–76.

[10] Khan SH, Sohail A, Khan A, Hassan M, Lee YS, Alam J, et al. COVID-19 detection in chest X-ray images using deep boosted hybrid learning. Comput Biol Med. 2021 Oct;137:104816.

[11] Kumar M, Shakya D, Kurup V, Suksatan W. COVID-19 prediction through X-ray images using transfer learning-based hybrid deep learning approach. Materials Today: Proceedings. 2021 Dec. https://linkinghub.elsevier.com/retrieve/pii/S2214785321078470.

[12] Manickam A, Jiang J, Zhou Y, Sagar A, Soundrapandiyan R, Samuel RDJ. Automated pneumonia detection on chest X-ray images: A deep learning approach with different optimizers and transfer learning architectures. Measurement J Int Measurement Confederation. 2021 Nov;184:109953.

[13] Guan Q, Huang Y, Zhong Z, Zheng Z, Zheng L, Yang Y. Thorax disease classification with attention guided convolutional neural network. Pattern Recognition Letters. 2020;131:38–45. doi: https://doi.org/10.1016/j.patrec.2019.11.040.

[14] Rahman M, Cao Y, Sun X, Li B, Hao Y. Deep pre-trained networks as a feature extractor with XGBoost to detect tuberculosis from chest X-ray. Comput Electr Eng. 2021 July;93:107252.

[15] Kora P, Ooi CP, Faust O, Raghavendra U, Gudigar A, Chan WY, et al. Transfer learning techniques for medical image analysis: A review. Biocybernetic Biomed Eng. 2022 Jan;42:79–107. https://linkinghub.elsevier.com/retrieve/pii/S0208521621001297.

[16] Masud M. A light-weight convolutional neural network architecture for classification of COVID-19 chest X-Ray images. Multimedia Syst. 2022 Jan;28:1165–74. https://link.springer.com/10.1007/s00530-021-00857-8.

[17] Ye Q, Xia J, Yang G. Explainable AI for COVID-19 CT classifiers: An initial comparison study. in: 2021 IEEE 34th International Symposium on Computer-Based Medical Systems (CBMS); 2021, p. 521–6.

[18] Codella NCF, Nguyen QB, Pankanti S, Gutman D, Helba B, Halpern A, et al. Deep learning ensembles for melanoma recognition in dermoscopy images 1. IBM J Res Develop. 2017;61:5-1.

[19] Alshazly H, Linse C, Barth E, Martinetz T. Ensembles of deep learning models and transfer learning for ear recognition. Sensors (Switzerland). 2019;19:1–26.

[20] Ayaz M, Shaukat F, Raja G. Ensemble learning based automatic detection of tuberculosis in chest X-ray images using hybrid feature descriptors. Phys Eng Sci Med. 2021;44:183–94. doi: https://doi.org/10.1007/s13246-020-00966-0.

[21] Dey S, Bhattacharya R, Malakar S, Mirjalili S, Sarkar R. Choquet fuzzy integral-based classifier ensemble technique for COVID-19 detection. Comput Biol Med. 2021 Aug;135:104585.

[22] Rajaraman S, Siegelman J, Alderson PO, Folio LS, Folio LR, Antani SK. Iteratively pruned deep learning ensembles for COVID-19 detection in chest X-Rays. IEEE Access. 2020;8:115041–50.

[23] Pitroda V, Fouda MM, Fadlullah ZM. An explainable AI model for interpretable lung disease classification. In: 2021 IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS); 2021. p. 98–103.

[24] Naz Z, Khan MUG, Saba T, Rehman A, Nobanee H, Bahaj SA. An explainable AI-enabled framework for interpreting pulmonary diseases from chest radiographs. Cancers. 2023;15(1):314. https://www.mdpi.com/2072-6694/15/1/314.

[25] Ribeiro MT, Singh S, Guestrin C. Why Should I. Trust You?: Explaining the Predictions of Any Classifier. 2016. https://arxiv.org/abs/1602.04938.

[26] Gong H, Wang M, Zhang H, Elahe MF, Jin M. An explainable AI approach for the rapid diagnosis of COVID-19 using ensemble learning algorithms. Front Public Health. 2022;10:874455. https://www.frontiersin.org/articles/10.3389/fpubh.2022.874455.

[27] Utomo A, Juniawan EF, Lioe V, Santika DD. Local features based deep learning for mammographic image classification: in comparison to CNN models. Procedia Comput Sci. 2021;179:169–76. 5th International Conference on Computer Science and Computational Intelligence 2020. https://www.sciencedirect.com/science/article/pii/S1877050920324649.

[28] Bhatt H, Shah M. A Convolutional neural network ensemble model for pneumonia detection using chest X-ray images. Healthcare Analytics. 2023 Nov;3:100176. https://linkinghub.elsevier.com/retrieve/pii/S2772442523000436.

[29] Althaqafi T, AL-Ghamdi ASAM, Ragab M. Artificial intelligence based COVID-19 detection and classification model on chest X-ray images. Healthcare. 2023 Apr;11:1204. https://www.mdpi.com/2227-9032/11/9/1204.

[30] Khanna M, Agarwal A, Singh LK, Thawkar S, Khanna A, Gupta D. Radiologist-level two novel and robust automated computer-aided prediction models for early detection of COVID-19 infection from chest X-ray images. Arab J Sci Eng. 2023:48:11051–83.

[31] Huy VTQ, Lin CM. An improved densenet deep neural network model for tuberculosis detection using chest X-ray images. IEEE Access. 2023;11:42839–49. https://ieeexplore.ieee.org/document/10108980/.

[32] Oh J, Park C, Lee H, Rim B, Kim Y, Hong M, et al. OView-AI supporter for classifying pneumonia, pneumothorax, tuberculosis, lung cancer chest X-ray images using multi-stage superpixels classification. Diagnostics. 2023 Apr;13:1519. https://www.mdpi.com/2075-4418/13/9/1519.

[33] Motwani A, Shukla PK, Pawar M, Kumar M, Ghosh U, Alnumay W, et al. Enhanced framework for COVID-19 prediction with computed tomography scan images using dense convolutional neural network and novel loss function. Comput Electr Eng. 2023 Jan;105:108479.

[34] Khanna M, Singh L, Thawkar S, Goyal M. Deep learning based computer-aided automatic prediction and grading system for diabetic retinopathy. Multimedia Tools Appl. 2023 Mar;82:1–48.

[35] Groen AM, Kraan R, Amirkhan SF, Daams JG, Maas M. A systematic review on the use of explainability in deep learning systems for computer aided diagnosis in radiology: Limited use of explainable AI? Eur J Radiol. 2022 Dec;157:110592. doi: 10.1016/j.ejrad.2022.110592.

[36] van der Velden BHM, Kuijf HJ, Gilhuijs KGA, Viergever MA. Explainable artificial intelligence (XAI) in deep learning-based medical image analysis. Med Image Analysis. 2022;79:102470. https://www.sciencedirect.com/science/article/pii/S1361841522001177.

[37] Zou L, Goh HL, Liew CJY, Quah JL, Gu GT, Chew JJ, et al. Ensemble image explainable AI (XAI) algorithm for severe community-acquired pneumonia and COVID-19 respiratory infections. IEEE Trans Artif Intelligence. 2023;4(2):242–54.

[38] Chaddad A, Peng J, Xu J, Bouridane A. Survey of explainable AI techniques in healthcare. Sensors. 2023;23(2):634. https://www.mdpi.com/1424-8220/23/2/634.

[39] Shen D, Wu G, Suk HII. Deep learning in medical image analysis. Annual Rev Biomed Eng. 2017 June;19:221–48. doi: 10.1146/annurev-bioeng-071516.

[40] Kinger S, Kulkarni V. Explainable AI for deep learning based disease detection. In: 2021 Thirteenth International Conference on Contemporary Computing (IC3-2021). IC3 '21. New York, NY, USA: Association for Computing Machinery; 2021. p. 209–16. doi: 10.1145/3474124.3474154.

[41] Khanna M, Singh LK, Thawkar S, Goyal M. PlaNet: a robust deep convolutional neural network model for plant leaves disease recognition. Multimed Tools Appl. 2023 May;1–53. doi: https://doi.org/10.1007/s11042-023-15809-9.

[42] Mahesh Kumar NB, Premalatha K, Suvitha S. Lung disease detection using self-attention generative adversarial capsule network optimized with sun flower optimization algorithm. Biomedi Signal Process Control. 2023;79:104241. https://www.sciencedirect.com/science/article/pii/S1746809422006954.

[43] Vats S, Sharma V, Singh K, Katti A, Mohd Ariffin M, Nazir Ahmad M, et al. Incremental learning-based cascaded model for detection and localization of tuberculosis from chest X-ray images. Expert Systems Appl. 2023;238:122129. https://www.sciencedirect.com/science/article/pii/S0957417423026313.

[44] de Almeida PAC, Borges DL. A deep unsupervised saliency model for lung segmentation in chest X-ray images. Biomed Signal Process Control. 2023;86:105334. https://www.sciencedirect.com/science/article/pii/S174680942300767X.

[45] Prasad Koyyada S, Singh TP. An explainable artificial intelligence model for identifying local indicators and detecting lung disease from chest X-ray images. Healthcare Analytics. 2023;4:100206. https://www.sciencedirect.com/science/article/pii/S2772442523000734.

[46] Lecun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc IEEE. 1998;86(11):2278–324.

[47] Fukushima K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol Cybernetics. 1980;36;193–202.

[48] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. in: 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings. 2015. http://www.robots.ox.ac.uk/.

[49] 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. 2016. p. 770–8. http://image-net.org/challenges/LSVRC/2015/.

[50] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2016. p. 2818–26.

[51] Huang G, Liu Z, Maaten LVD, Weinberger KQ. Densely connected convolutional networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2017. p. 2261–9. https://github.com/liuzhuang13/DenseNet.

[52] Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. Grad-CAM: Visual explanations from deep networks via gradient-based localization. Int J Comput Vision. 2019 Oct;128(2):336–59. https://doi.org/10.1007.

[53] Simonyan K, Vedaldi A, Zisserman A. Deep inside convolutional networks: visualising image classification models and saliency maps. 2013. https://arxiv.org/abs/1312.6034.

[54] Friedman JH. Greedy function approximation: a gradient boosting machine. Annals Stat. 2001:1189–232.

[55] Apley DW, Zhu J. Visualizing the effects of predictor variables in black box supervised learning models. J R Stat Soc Ser B Stat Methodol. 2020;82(4):1059–86.

[56] Alex Goldstein JB, Kapelner A, Bleich J, Pitkin E. Peeking inside the black box: visualizing statistical learning with plots of individual conditional expectation. J Comput Graph Stat. 2015;24(1):44–65. doi: https://doi.org/10.1080/10618600.2014.907095.

[57] Lundberg SM, Lee SI. A unified approach to interpreting model predictions. Adv Neural Inform Process Syst. 2017;30. ISBN: 9781510860964.

[58] Fan A, Jernite Y, Perez E, Grangier D, Weston J, Auli M. ELI5: long form question answering; 2019.

[59] Hu X, Chu L, Pei J, Liu W, Bian J. Model complexity of deep learning: a survey; 2021.

[60] Shah B, Bhavsar H. Time complexity in deep learning models. Proc Comput Sci. 2022;215:202–10. 4th International Conference on Innovative Data Communication Technology and Application. https://www.sciencedirect.com/science/article/pii/S1877050922020944.

[61] Niu S, Liu Y, Wang J, Song H. A decade survey of transfer learning (2010–2020). IEEE Trans Artif Intell. 2020;1(2):151–66.

[62] Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, et al. MobileNets: efficient convolutional neural networks for mobile vision applications; 2017. doi: 10.48550/ARXIV.1704.04861.

[63] Chollet F. Xception: "Deep Learning with Depthwise Separable Convolutions," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, pp. 1800–1807, doi: 10.1109/CVPR.2017.195.

[64] Tan M, Le QV. EfficientNet: rethinking model scaling for convolutional neural networks. In K. Chaudhuri & R. Salakhutdinov, editors, Proceedings of the 36th International Conference on Machine Learning; 2019 (pp. 6105–6114). Retrieved from https://proceedings.mlr.press/v97/tan19a.html.

[65] Woo S, Debnath S, Hu R, Chen X, Liu Z, Kweon IS, et al. ConvNeXt V2: Co-designing and Scaling ConvNets with Masked Autoencoders; 2023.

[66] Prasad Koyyada S, Singh TP. A multi stage approach to handle class imbalance: An ensemble method. Proc Comput Sci. 2023;218:2666–74. International Conference on Machine Learning and Data Engineering. https://www.sciencedirect.com/science/article/pii/S1877050923002399.

[67] Ronneberger O, Fischer P, Brox T. U-Net: convolutional networks for biomedical image segmentation. In: Navab N, Hornegger J, Wells W, Frangi A, editors. Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015. MICCAI 2015. Lecture Notes in Computer Science(), vol. 9351. Springer, Cham. doi: 10.1007/978-3-319-24574-4_28.