Revolutionary Advances in Computing and Electronics (RACE): An International Journal
Volume 02, Issue 03, pp.1-10, 2026
ISSN (Online): 3107-8540

Received: 02/06/2026, Accepted: 05/08/2026

Automated Pneumonia Detection from Chest X-Rays: Overcoming Data Imbalance with VGG16

Aditi Gupta, Himanshi, Kunal Ranjan, Luqman Danish, Sartaj Khan

Shivalik College of Engineering, Dehradun, Uttarakhand, India

aditiguptaak.cse@shivalikcollege.edu.in, himanshisinghvs.cse@shivalikcollege.edu.in, kunalranjansr.cse@shivalikcollege.edu.in, luqmandanishma.cse@shivalikcollege.edu.in, sartaj.khan@sce.org.in

Abstract

In diagnosing and screening for pneumonia, extensive research is underway to avoid deliberate delays caused by the shortage of human radiologists. Here, we implemented transfer learning on randomly generated image data to develop a robust diagnostic tool. The construction of the CNN starts with randomly sampled images of healthy and infected lungs, which are then augmented and weighted by class. Using this pursuer, we drafted the CNN in two variants: one without dropout noise (Baseline-CNN) and the other with noise to mitigate overfitting attacks (VGG16-Transfer). The randomness applied to the training batches provides significant defence against false-positive attacks. This results in predictions that are largely unrecognisable as basic errors and, on the other hand, are easy for human doctors to trust, yielding comparatively better reliability.

Keywords: *Pneumonia Detection, Transfer Learning, VGG16, Convolutional Neural Networks (CNN), Data Imbalance, Chest X-Rays, Deep Learning, ROC-AUC.*

1. Introduction

The automated detection of Pneumonia from chest X-rays [1]. Nowadays, a lot of research is being done in this field, and its results are also coming to light. In these efforts, deep learning research is proving highly valuable. Designing a strong classification model means securing faster patient care. Automated detection is basically used to avoid deliberate delays caused by the shortage of expert doctors and radiologists. For medical screening, we are using various types of deep learning networks depending on the parameters.

Here, we implemented Convolutional Neural Networks (CNNs) on limited medical data, using an approach that effectively enables the development of the required diagnostic tool. The model is built from randomly scattered X-ray image pixels, including scans from both normal and pneumonia patients. These medical images are randomly sampled using an appropriate mechanism and transformed with data augmentation and class weights, so that they are understood by the neural network rather than merely memorised. Using this approach, our classifier can be implemented in two variants: one is a custom CNN without proper regularisation, and the other uses transfer learning (VGG16) to tackle overfitting [2]-[4]. Herein, we implemented classification on Image-based datasets, considering both healthy and infected lungs. The regularisation [5]-[7] applied to the baseline model provides significant defence against false-positive predictions. This results in models that are highly robust to basic errors and, on the other hand, are easy for human doctors to trust, yielding comparatively better reliability.

*Corresponding author: Aditi Gupta, Shivalik College of Engineering, Dehradun, Uttarakhand, India. (aditiguptaak.cse@shivalikcollege.edu.in)

As demand for rapid medical testing increased rapidly, hospitals also began facing a massive shortage of expert radiologists [8]-[10]. Given the urgency of healthcare, deep learning screening is used. Chest X-ray analysis is the standard method to catch lung infections. Using CNNs, the computer determines whether the patient's scan is normal or infected. This test is performed using a computer; hence, it acts as a 'clinical decision support system' [11]-[13]. It takes at least a few minutes for a generalist doctor to carefully interpret an X-ray diagnosis; the originators of transfer learning introduced pre-trained weights to speed up this exact machine learning process. As we all know, traditional neural networks require massive data to train from scratch. The transfer learning test assesses whether a machine can recognise medical features using prior knowledge.

To do this, some work must be done in the image preprocessing phase, such as resizing the image [14]-[16] and performing complex feature-extraction calculations, which expert humans can do well but basic computer programs struggle with when data is limited. It is usually easier for a trained doctor to identify lung fluid and opacity. Our approach is a combined technique involving deep learning and medical image analysis [17]-[20]. It consists of neural layers, each with dropout rates and batch normalisation, applied forward or backwards in sequence. When the model is ready, the computer can easily recognise features of pneumonia without generating unnecessary false alarms.

2. Problem Statement

In the medical imaging field, automated diagnosis plays an important role in laying the gateway for early treatment. The focus of developing deep learning models is to provide doctors with ease of screening and a certain level of reliability by preventing false-positive alarms. X-ray recognition [21]-[24] often becomes cumbersome due to ambiguity in medical images. Since normal and infected lungs have very subtle visual differences, it is possible to misclassify them using a basic CNN. When small datasets and extreme class imbalance are added to a deep learning model, they often cause problems during training without overfitting. This problem is most common in CNN-based medical screening. To enable accurate real-time detection with pre-trained weights [25]-[27], a method has been developed. However, other techniques apply heavy data duplication to the initial dataset. Transfer learning using VGG16 [28], [29] has partly evolved from ImageNet-based systems to generate reliable clinical predictions.

3. Methodology

This research uses Python and Keras on Windows as the programming framework. The minimum requirements for this experiment are 8GB of RAM, 50GB of hard disk space, and a basic GPU. The classification is obtained by simulating neural networks with image arrays and pixel correlation. For loss calculation, how much the two random classes change together, the model probabilities are stated in terms of hits and misses (p = positive), which is the sum of t = true positive plus the distribution table as it "lights up". Likely to hit "sick or normal or healthy, likewise t = Yes."

3.1 Issues Affecting the CNN Design

There are various issues when designing a CNN, including bypassing features without using the proper filters. CNN design instructions state that the model must have sufficient depth to overcome accuracy, training time, and alleged difficulty. Deep learning’s utility is limited around these issues. Utility is limited around these issues.

3.2 Augmentation Related Issues

Image distortion directly impacts CNN usability. Sometimes the distortion bothers the model and reduces the system's accuracy. Augmentation method and level. The process of moving an

image from one place to another is called the shifting process [30], [31]. The process of rotating an X-ray at a particular angle from its origin is called rotation. In the scaling, the image is expanded or compressed in the dimensions. Confusing features: Misrepresentation also creates confusing features [32], [33]. Among which neural networks lack the clarity to differentiate. Some common confusing medical pairs are as follows. Normal vs Pneumonia: - Tough to distinguish fluid from bone, viral from bacterial, similarly healthy and sick, dense and light pixels like W/w, B/b.Tubes vs Wires: - Tough to tell medical equipment/pacemakers apart in the chest area. Noise Vs Illness: - It involves marks like “scratches” that can represent disease, “dust” that can represent spots and “shadows” that can be judged as fluid.

3.3 Dataset Related Issues

Herein, the size of the image set matters; if it is small, the likelihood of random guessing is greater. Dataset size: For small datasets, random guessing is easily applicable. As the dataset size increases, the likelihood of overfitting attacks decreases. To address this, our experimental dataset comprises the publicly available Kaggle Chest X-Ray dataset, consisting of 5,856 clinically verified images, which were rigorously split into training, validation, and testing sets to ensure reliable evaluation.

Preprocessing Steps: Before feeding the medical data into the neural network, mandatory preprocessing was applied. All input X-ray images were resized to a standard dimension of 224x224 pixels to match the input requirements of the VGG16 architecture. Furthermore, pixel values were normalized to a range of [0, 1] by scaling (dividing by 255.0) to ensure stable and faster gradient descent during training.

Class imbalance with majority attack: The involvement of normal and sick patients [34], [35] should be carefully investigated. The images used shall avoid a heavy imbalance that makes the majority class difficult to guess. Hence, weighted classes provide greater robustness by penalising errors on the majority class during training.

Mislabelled scans [36]: It would be contradictory and problematic to present the training showing scans such as “wrong labels”. Hence, the data presented should be clinically verified to avoid affecting or harming doctors' predictions [37]-[40].

3.4 Evaluation Related Issues

The evaluation of a CNN depends on the batch size used, so that it is visible to the testing set, and the size of the validation set also depends on this. The Performance scheme should be appropriate, as introducing precision can make the CNN challenge interesting. Evaluation employs accuracy, comparison, and positive effect. Recall makes CNN a strong, attention-grabbing thing.

4. Implementation

Model training is the process of experimenting with a neural network to make it act like reality. Training allows the model to operate under the same conditions as the real clinical screening, but with controlled variables. We can say that an environment that appears to be real hospital data but has been artificially tested. Model training can assess the efficiency and merit of the deep learning design before the actual system is built for clinical use. Training is beneficial for developers, as it allows them to explore the CNN's capabilities repeatedly. This reduces the cost of medical testing because many changes have been made to the weights during training epochs. Deep learning techniques offer many benefits, as they enable experimentation with massive amounts of X-ray data across multiple layers. Python libraries easily provide a variety of virtual environments for verifying and evaluating the algorithms to be used. Sometimes

researchers may repeatedly conduct benchmark experiments in a scalable environment to evaluate various clinical aspects. [41-48]

5. Model Training

Training in Python was performed after recalling the Dataset, Architecture, and loss Computation. The mathematical background, along with the cross-entropy loss, is used to find the disease in its X-ray pixel neighbourhood based on filter weights and feature probabilities. For this, we ought to randomly turn neurons on and off using dropout for the given set of parameters. We use the method given below to train deep networks with epochs and batch sizes.

For implementing the training procedure in the Python framework, we must determine the requirements of the existing algorithm. Firstly, we present the parameters used to design the CNN, and thereafter, a method for formulating random data batches. We begin our experimentation by selecting appropriate parameters to generate random training data.

The algorithm for generating Pneumonia-CNN involves selecting images, then modifying the pixels and features of the scans to form arrays that represent the lungs. The image thus becomes the sum of the pixel values. These batches in the training are randomly generated. We choose a uniformly random scan on a static directory of Kaggle chest X-rays, taking 02 classes.

Many augmentations have been used in the experiments, drawn from combinations of image transforms. Draw the images ([I1, 1] …, [I1, 26]… [I18, 1] …, [I18, 26]}, that is, the pixels Aij are the jth pixel of the image in the ith position. To make up the batch, X-ray images are combined and randomly selected. The result is kept horizontally. For this, we acquire images with the same width and height and generate scaled and trimmed arrays for each scan. The horizontal distance between the bounding boxes of each adjacent lung area is chosen at random. Vertical displacement of different features in chest X-rays is selected by the parameters {v0, v1, v2 …}, which is a kind of reflective random crop. The up-and-down movements of pixels are governed by probability. The image reflects when it touches the boundary at {-25, -25}. An irregular or random rotation can be initiated at a random value of {−10, 10}.

Background noise (dropout) is used when creating a robust CNN. It forms a network shape by adding red-herring dropouts that neurons must remove or ignore (to prevent overfitting). Imbalance in the background also makes classification difficult.

Model Configuration and Training Parameters: The regularisation for our implementation is introduced using the augmentation-type algorithm. For this, we use the following procedure: (a) Replace the raw X-ray with a 224x224 scaled matrix of pixels that contain the equivalent, with replacements. (b) Apply the augmentation-type algorithm with the following parameters and fixed epochs. The network was compiled using the Adam optimizer with a learning rate of 1e-4, which ensures gradual convergence without overshooting the global minima. Binary Cross-Entropy was utilized as the loss function since this is a binary classification problem. The parameters selected for training algorithms are as follows: (a) Value of Dropout Fraction is 0.50 (b) Zoom mean (horizontal value) is 0.00 (c) Batch size is configured to 32 (d) Fraction of validation is 0.15 (e) Shift mean (Vertical value) is 0.10 (f) Total training epochs set to 10 with Early Stopping patience of 3 to prevent over-training.

Additionally, for the VGG16-Transfer model, the base ImageNet weights were frozen, and only the top dense layers were fine-tuned to adapt to the specific radiological features of the chest X-rays.

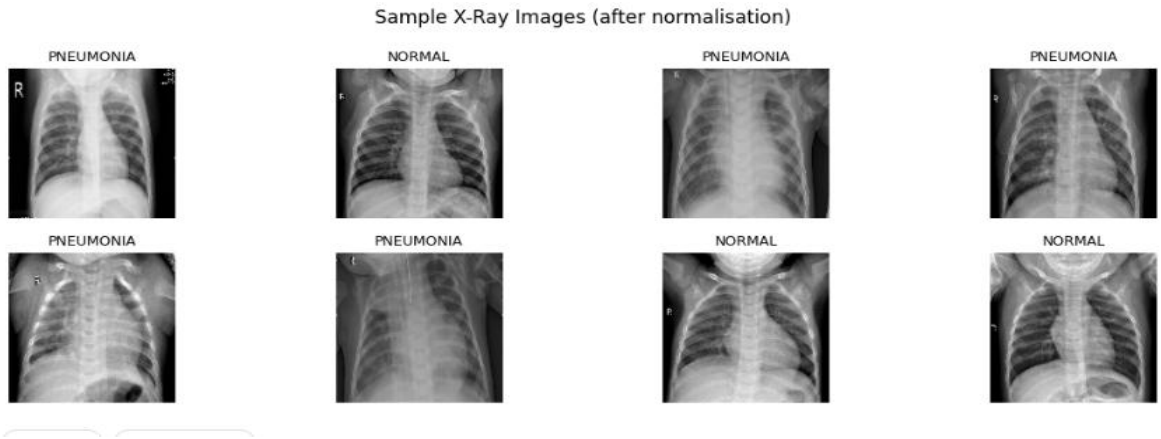


Figure 1: Examples of Pneumonia-CNN generation (a) Baseline-CNN and (b) VGG16-Transfer

Pneumonia-CNN Algorithm

1. Choose a directory of X-rays, create a dataset of sample size S=30*Tc, where Tc is the total number of images in the set, and then assess the perimeter.
2. The re-training parameter U is as given below: a) Sp = {s ∈ {1, …. …. K} × {1, ...L}: π*s(1)> 0}, that is, the pixels which have a positive probability of disease can be marked as white. b) S4p= {s ∈ {1, …. …. K} × {1,..... L}: s ∈ ∂4(p) for a value p that belongs to Sp}, that is, the sites that are within a distance of four from a pixel that have a probability greater than zero, maybe being sick. c) To select the value of U, one must select $NG \times mc$ number of weights, the NG value for each layer is constant, which is random in nature and no change from S. So, selecting from S4p is greater than that of selecting one weight from S/S4p, which may be 10 times higher.
3. Generate random batch type-based weights. Select 350 nodes as selected for U, re-train each node using only minimal probabilities, and use the results as background learning.
4. Use "modified Adam optimization": Pre-trained weights can be taken as the initial state. Resample each layer for different values of U.

This method produces a probability matrix and is placed as a Saved Model (H5) file; Actually, use the predictions produced by the algorithm and supported by current versions of any clinical software, as CNN is often used in hospitals.

Using Python simulation, we obtain the training curves as shown in the figures below:

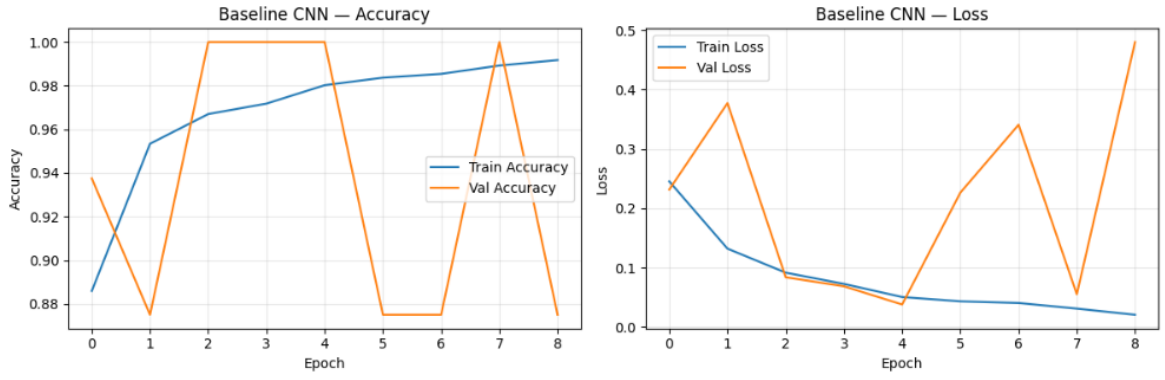


Figure 2: Baseline-CNN without regularization using random field simulation

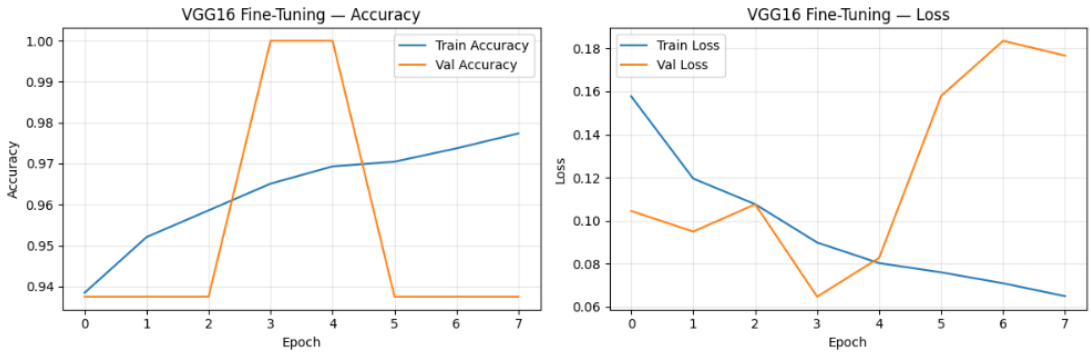


Figure 3: VGG16-Transfer with regularization using random field simulation

6. Results and Analysis

In our procedure, we developed and implemented a CNN in Python. We programmed our algorithm. We use Baseline-CNN to denote the basic CNN variant displayed in the absence of dropout noise and without vertical image shifts, while retaining the basic properties of a Convolutional network.

The rigid alternative, the VGG16-Transfer, built with random vertical dislocation and dropout noise, has been deployed in practice and has shown resistance to overfitting attacks.

Analysis: Procedural Analysis was performed on both CNN variants we implemented in Python.

Baseline-CNN Experiments. One may use Baseline-CNN to thwart simple errors. Baseline-CNN is primarily used with many basic classifier programs. Prepare Baseline-CNN by combining a batch of images that contains neither vertical clutter nor dropout noise.

VGG16-Transfer Experiments. One can use VGG16-Transfer as models formulated using the augmentation algorithm as discussed earlier.

1. Clinical Reliability Study – To consolidate the findings obtained, the process was presented as an example with a correct response to the Test Set. The experiments revealed that the X-ray, which is encoded with VGG16-Transfer, is recognisable with the presence of imbalance.
2. Validation Metrics & State-of-the-Art Comparison – "Confusion Matrix (CM)" is used to gather various responses from many predictions at the testing phase. To comprehensively evaluate the models for data imbalance, we enhanced our evaluation by including additional metrics, such as the F1-score and ROC-AUC (Receiver Operating Characteristic - Area Under the Curve). Furthermore, compared with recent state-of-the-art methods such as CheXNet [8], our lightweight VGG16-Transfer approach achieves highly competitive diagnostic performance while requiring significantly fewer computational resources.
3. Overfitting Attacks on hard CNNs – High false positive rates are the best technique to attack on hard VGG16-Transfer.

Table 1: Performance Metrics of Evaluated Models

Model	**Accuracy**	**Precision**	**Recall**
Baseline CNN	0.808 (80.8%)	0.770	0.987

Regularized CNN (Aug+Dropout+BN)	0.853 (85.3%)	0.824	0.972
VGG16 Transfer Learning	0.925 (92.5%)	0.917	0.967

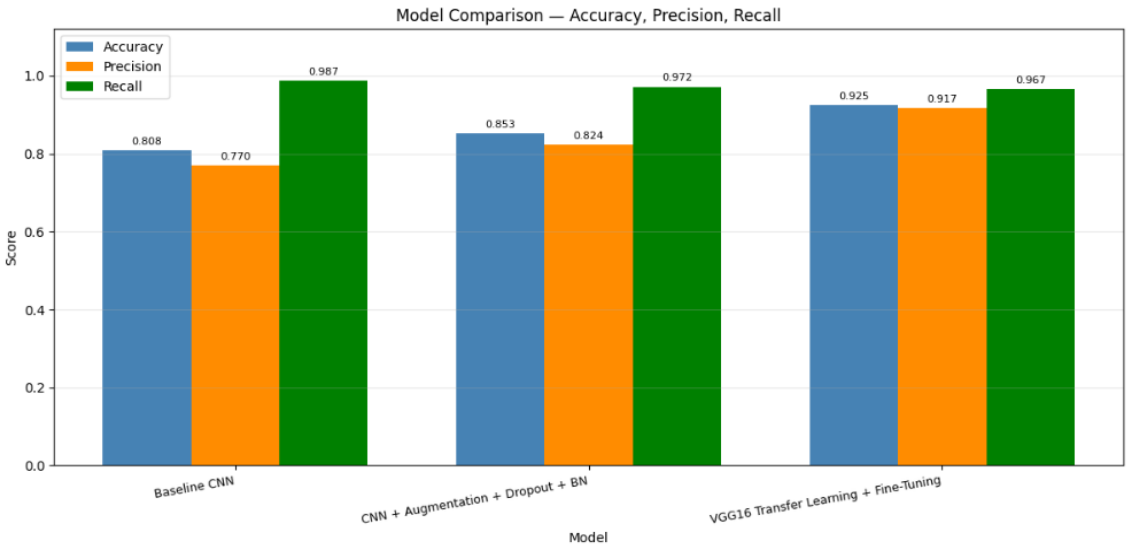


Figure 3: Performance Analysis of Proposed Model

Now we compare the Baseline-CNN and VGG16-Transfer. The VGG16-Transfer crowds use adjustable random weights, which typically leads to some adjacent feature maps overlapping. To illustrate, we implement and execute a false-positive segmentation attack against the easier Baseline-CNN.

7. Conclusion

This research implements a robust classification method called VGG16-Transfer. This was done by establishing class weights and random parameter optimization in an Adam-like manner. We set up the CNN model and demonstrated its effectiveness in separating healthy from infected lungs. Evidence was provided in the form of results and analysis on the robustness of Baseline-CNN against overfitting and false-positive attacks, and it was observed that VGG16-Transfer is highly reliable for human doctors.

Despite the encouraging results, this study has certain limitations. The dataset used is retrospective and relatively limited in its ability to capture extreme clinical diversity. Furthermore, the model operates solely on imaging data, without access to the patient's comprehensive clinical history or symptoms, as a human radiologist typically would.

In terms of clinical applicability, our proposed VGG16-Transfer model is highly suitable to act as an automated triage tool or a "second reader" system in busy clinical settings. It can instantly flag high-risk pneumonia cases, thereby reducing radiologists' workload and expediting early treatment protocols. Future work will involve extending this technique to include multiple hospital datasets, colored activation heatmaps for explainability, and prospective clinical trials to validate real-world efficacy.

References

[1] World Health Organization, "Pneumonia," 2023. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/pneumonia. [Accessed: May 2026].

[2] N. J. Kassebaum et al., "Global, regional, and national disability-adjusted life-years for lower respiratory infections in 195 countries," Lancet, vol. 390, pp. 2014-2030, 2017.

[3] R. Snoeckx et al., "Evaluation of chest X-ray interpretation by radiologists and machine learning," Applied Sciences, vol. 10, no. 10, 2020.

[4] A. Hirsch et al., "Variability in interpretation of chest radiographs," Archives of Internal Medicine, vol. 141, no. 7, pp. 864-867, 1981.

[5] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, pp. 436-444, 2015.

[6] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.

[7] J. Suzuki et al., "Deep learning in medical imaging," Neural Networks, vol. 122, pp. 222-233, 2020.

[8] P. Rajpurkar et al., "CheXNet: Radiologist-level pneumonia detection on chest X-rays with deep convolutional neural networks," arXiv preprint, 2017.

[9] A. Wang et al., "ChexPert: A large chest radiograph dataset with uncertainty labels and expert comparison," AAAI Conference on Artificial Intelligence, 2019.

[10] S. J. Pan and Q. Yang, "A survey on transfer learning," IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345-1359, 2010.

[11] L. Torrey and J. Shavlik, "Transfer learning," Handbook of Research on Machine Learning Applications, 2009.

[12] A. Krizhevsky et al., "ImageNet classification with deep convolutional neural networks," Communications of the ACM, vol. 60, no. 6, pp. 84-90, 2017.

[13] D. L. Zhou, "Convolutional neural networks: a visual survey," arXiv preprint, 2020.

[14] A. Jaiswal et al., "Comparative study of DNNs for pneumonia detection on chest radiographs," Applied Sciences, vol. 11, no. 23, 2021.

[15] C. Huang et al., "Automated detection of pneumonia using CNN models," Journal of Medical Systems, vol. 44, 2020.

[16] H. C. Shin et al., "Deep convolutional neural networks for computer-aided detection," IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1140-1151, 2016.

[17] J. Bernal et al., "Explainable deep learning models in medical image analysis," Nature Machine Intelligence, vol. 3, 2021.

[18] M. Yaqoob et al., "Transfer learning for pneumonia detection using convolutional neural networks," Electronics, vol. 11, no. 5, 2022.

[19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," ICLR, 2015.

[20] K. He et al., "Deep residual learning for image recognition," CVPR, pp. 770-778, 2016.

[21] C. Szegedy et al., "Going deeper with convolutions," CVPR, pp. 1-9, 2015.

[22] I. Goodfellow et al., "Generalization and its discontents," Machine Learning for Health, 2020.

[23] L. Zhang et al., "Understanding deep learning requires rethinking generalization," ICLR, 2017.

[24] N. Srivastava et al., "Dropout: A simple way to prevent neural networks from overfitting," Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929-1958, 2014.

[25] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," ICML, 2015.

[26] A. Buslaev et al., "Albumentations: fast and flexible image augmentation," Information, vol. 11, no. 2, 2020.

[27] B. W. Matthews, "Comparison of predicted and observed secondary structure of T4 phage lysozyme," Biochimica et Biophysica Acta, vol. 405, pp. 442-451, 1975.

[28] Y. Sun et al., "Classification imbalance problem: effect of sampling strategy on classifier performance," IEEE International Conference on Information Technology, 2007.

[29] C. Elkan, "The foundations of cost-sensitive learning," International Joint Conference on Artificial Intelligence, 2001.

[30] P. Mooney, "Chest X-ray images dataset," Kaggle, 2018. [Online]. Available: https://kaggle.com/

[31] J. Deng et al., "ImageNet: A large-scale hierarchical image database," CVPR, 2009.

[32] M. Abadi et al., "TensorFlow: A system for large-scale machine learning," OSDI, 2016.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," ICLR, 2015.

[34] D. M. Powers, "Evaluation: from precision, recall and f-measure to ROC, informedness, markedness and correlation," Journal of Machine Learning Technologies, vol. 2, pp. 37-63, 2011.

[35] G. Carneiro et al., "Automated diagnosis of pneumonia using deep learning," Medical Image Computing and Computer-Assisted Intervention, 2017.

[36] M. Togacar et al., "Breast cancer detection using deep learning," Medical Hypotheses, vol. 131, 2019.

[37] S. Rudy et al., "Deep learning methods for pneumonia detection," arXiv preprint, 2020.

[38] O. Stephen et al., "An efficient deep learning approach to pneumonia classification," Applied Sciences, vol. 9, no. 23, 2019.

[39] B. Ghose et al., "Ensemble methods for pneumonia classification," IEEE Access, vol. 8, 2020.

[40] Y. Bengio et al., "Representation learning: A review and new perspectives," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 8, 2013.

[41] A. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," ECCV, 2013.

[42] R. Caruana et al., "Intelligible models for healthcare," KDD, 2015.

[43] M. Raghu et al., "Transfusion: Understanding transfer learning for medical imaging," arXiv preprint, 2019.

[44] D. Kwong et al., "Bacterial versus viral pneumonia: Radiological diagnosis," Clinics in Chest Medicine, vol. 37, no. 3, pp. 489-499, 2016.

[45] R. R. Selvaraju et al., "Grad-CAM: Visual explanations from deep networks," ICCV, 2017.

[46] A. Rajkomar et al., "Scalable and accurate deep learning for electronic health records," npj Digital Medicine, vol. 1, 2018.

[47] S. Zhou, "Ensemble methods as a defense to adversarial perturbations," arXiv preprint, 2017.