

# Adaptive Lung Diseases Images Classification Technique Based on Deep Learning



Nguyen Huu The, Nguyen Thi Hong Nhung, and Nguyen Thanh Binh

**Abstract** Medical images have made an important contribution to improving the accuracy and effectiveness of disease diagnosis, such as diseases related to lung, heart, liver, kidney, etc. Pneumonia has increased rapidly in the world in recent years. Chest X-ray image analysis is a common method for detecting lung diseases. An advanced artificial intelligence system will help doctors have accurate conclusions, timely treatment for patients and reducing mortality. Using machine learning on X-ray images is of great interest, but research results are still limited in accuracy. This paper proposed an adaptive technique for lung diseases image classification based on the deep learning method. We improved the convolutional neural network for lung diseases image classification, created a training model with a suitable number of hidden network layers and optimal algorithms to detect pneumonia images. As a result, the rate of correct detection of pneumonia image was 98.72%. We used chest X-ray images dataset that published by Kaggle, including 5863 chest X-ray images. The results of the proposed method are better than the other methods.

**Keywords** Pneumonia · Lung diseases · Convolutional neural network · Chest X-ray pneumonia

---

N. H. The · N. T. H. Nhung  
Faculty of Information Technology-Communication, University of Cuu Long, Vinh Long, Viet Nam  
e-mail: [nguyenhuuthe@mku.edu.vn](mailto:nguyenhuuthe@mku.edu.vn)

N. T. H. Nhung  
e-mail: [nguyenthihongnhung@mku.edu.vn](mailto:nguyenthihongnhung@mku.edu.vn)

N. T. Binh (✉)  
Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, VNU-HCM, 268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Viet Nam  
e-mail: [ntbinh@hcmut.edu.vn](mailto:ntbinh@hcmut.edu.vn)

Vietnam National University Ho Chi Minh City, Linh Trung Ward, Thu Duc District, Ho Chi Minh City, Viet Nam

# 1 Introduction

Detect pneumonia on chest X-rays is a challenging task even for X-ray specialists. Pneumonia on X-rays is difficult to distinguish from some pathologies about the other chest area [1]. This work requires knowledge of chest-related pathologies, as well as expertise and experience in viewing X-rays to make an initial diagnosis. The intelligent computer systems to support doctors became essential to help doctors diagnose pneumonia from X-ray images more effectively.

Many recent researches in the field of image processing have achieved good results in image classification, especially using Convolutional Neural Networks (CNN), a deep learning model that is heavily used in image recognition, because CNN is capable of extract key features from image fast and efficiently [2]. In the field of medicine, there have been many scientific papers that have applied the CNN model [3–9] such as breast cancer detection [3], skin cancer [4], pneumonia detection [9], etc.

To assist doctors in detecting pneumonia, Ayan [10] studied the diagnosis of pneumonia with the CNN model when tested on pretrained models of Xception and Vgg16. To adjust the parameters for the model in the process of training the model, the authors combined the method of transfer learning and fine-tuning. The result was the accuracy of Xception with 82% and Vgg16 with 87%. Ponnada [11] presented a system for detecting pneumonia. The author and collaborators suggested a CNN method consisting of seven hidden layers to train their model. The result with accuracy is 86%, precision is 79%, and recall is 98% with LIDC-IDRI and Mendeley dataset. Kadam [12] have developed a deep neural network, which helped predict the presence of pneumonia using chest X-rays, a CNN model has been employed to increase efficiency and accuracy, to further add to the performance, optimum differential learning rates have been selected using the techniques of cosine annealing and stochastic gradient with restarts, and the result with accuracy is 92.90%, precision is 90.88%, and recall is 99.27%.

The detection of pneumonia on chest X-ray is an image classification problem to determine whether the presence of pneumonia in the chest X-ray image. The system input is a chest X-ray image; the output is a normal X-ray image or pneumonia image. In this paper, we proposed an adaptive technique for lung diseases image classification based on the deep learning method. We improved the convolutional neural network for lung diseases image classification, created a training model with a suitable number of hidden network layers and optimal algorithms to detect pneumonia images. The rest of this paper is organized as following: in Sect. 2, we describe deep learning for classification. The proposed method for lung diseases image classification is presented clearly in Sect. 3. The experiment results and conclusion are presented in Sects. 4 and 5.

## 2 Background of Deep Learning for Classification

Deep learning is the neural network with multiple hidden layers. A neural network may have few hidden layers, but with deep learning, the number of hidden layers is very large, up to hundreds of layers. More layers will increase the accuracy of the neural network. In deep learning technique, the more the input data, the more accurate the output [5]. Deep learning is a sub-branch of machine learning; deep learning algorithms will perform a task several times, each time adjusting it a little to improve results.

CNN is a class in deep learning. CNN has applications in image and video identification [13], recommendation systems [14], image classification, medical image analysis [10–12], natural language processing [15], and financial time series [16]. CNN performs images classification by receiving input images, processing them, and classifying them as labels. The computer receives the input as an array of pixels based on the resolution of the image, from which the computer recognizes the image as height × width × dimension (h × w × d). An image with (150 × 150 × 3) means there are three color channels RGB.

In the CNN model, each input image scans through convolutional layers, also known as hidden layers. Each convolutional layer includes filters (kernel), activation function, max pooling, and the result is the feature map. To classify the output data, the feature map will switch to the fully connected layer (FC) and apply an activation function.

Figure 1 illustrates the process of receiving input image, extracting features of the image through layers, and classifying the output object.

The convolutional layer is used to extract features of the image. The input image of each convolutional layer is a 3-dimensional tensor of size (H × W × D) that will slide over by a kernel of size (Wk × Hk × Dk) with stride S and padding P. Each convolutional layer has N kernel; the output is a 3-dimensional tensor with the formula shown in the lecture [17]:

$$\left(\frac{W - W_k + 2P}{S} + 1\right) * \left(\frac{H - H_k + 2P}{S} + 1\right) * N \tag{1}$$

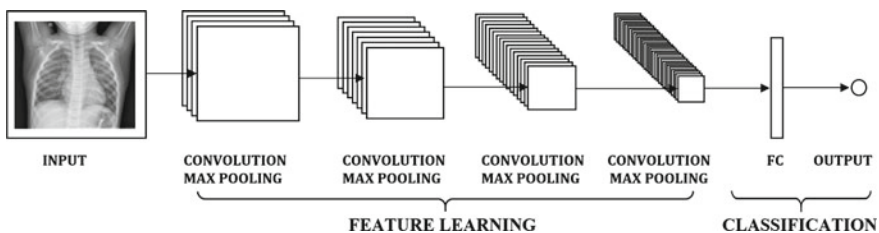


Fig. 1 A neural network with many convolutional layers

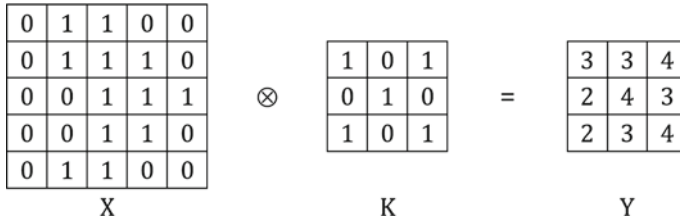


Fig. 2 Features map

Consider a matrix X [5 × 5] containing image pixels with values 0 or 1, and a filter K [3 × 3] slides up to X with stride = 1. Each time K slides over X, extract from X a matrix A [3 × 3], perform the element-wise calculation on two matrices by the formula:

$$Y = X \otimes K \tag{2}$$

For each element  $x_{ij}$  in a matrix X, take out a matrix equal to the size of kernel K, and element  $x_{ij}$  in the central, called matrix A. Then, sum the elements of the element-wise calculation of matrix A and matrix K, write the result into matrix Y. Figure 2 illustrates the element-wise calculation to get the features map [18].

The value of  $y_{11}$  is calculated:  $y_{11} = \text{sum}(A \otimes K) = x_{11} \times k_{11} + x_{12} \times k_{12} + x_{13} \times k_{13} + x_{21} \times k_{21} + x_{22} \times k_{22} + x_{23} \times k_{23} + x_{31} \times k_{31} + x_{32} \times k_{32} + x_{33} \times k_{33} = 3$ . And perform the same calculation with the remaining elements in the matrix.

However, color images have three color channels: red, green, blue. The image is represented as a 3-dimensional tensor, and the filter is also a 3-dimensional tensor of size  $(k \times k \times 3)$ .

The output of the convolutional layer will apply the activation function, and the input of the next convolutional layer will be received from the output of the previous layer as in the lecture [17]. Each kernel has dimensions  $(W_k \times H_k \times D_k)$  and has a bias coefficient, the total parameter of a kernel is  $(W_k \times H_k \times D_k + 1)$ . With N kernels in the layer, the total number of parameters in this layer is  $(N \times (W_k \times H_k \times D_k))$ .

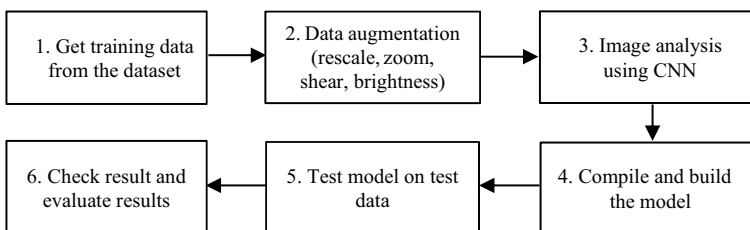
Pooling layers are used between convolutional layers, to reduce data size, but retain important attributes. The reduced data size helps reduce computations in the model. If the pooling layer is size =  $(2 \times 2)$ , stride = 2, padding = 0, the output size of width and height will be halved, the depth will not change. After the image has been transferred to many convolutional layers and pooling layers, the model has learned the features of the image. The last layer, the tensor of size  $(H \times W \times D)$ , will be converted to a size vector  $(H \times W \times D)$  in the fully connected layer. Finally, we combine the fully connected layer with image features to get the model’s output.

### 3 Lung Diseases Image Classification

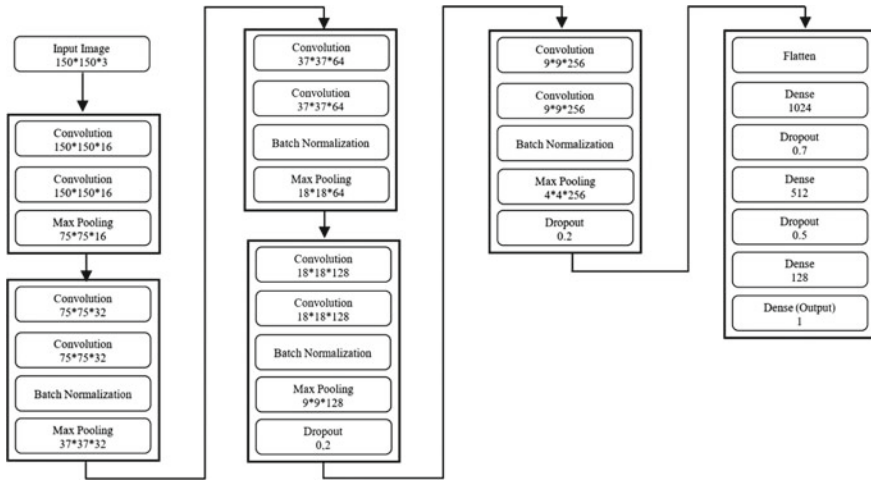
In this section, we proposed the method for lung diseases images classification based on deep learning. The proposed method is presented in Fig. 3.

In Fig. 3, the proposed method includes six steps. The detail of the above steps presents as follows:

1. *Get training data from the dataset*: perform reading chest X-ray image data from dataset “train” directory. Images, in turn, are put into the system, the total number of photos in training is 5218. The parameters are configured as follows,  $batch\_size = 32$ ,  $epoch = 32$ ,  $image\_dimension = 150 \times 150$  px. Therefore, corresponding to one epoch, the training dataset will be put into the system with 163 batches ( $5218/32$   $batch\_size$ ).
2. *Data augmentation*: deep learning techniques require a huge amount of data to get good results. In many cases, the amount of data is not large enough, such as medical data. In machine learning, the solution often chosen to solve this problem is data augmentation, help avoid redundancy and improve algorithm accuracy [19]. In this work, we have used different data augmentation techniques such as rescaling, image change adjust brightness, shear intensity, range for random zoom ( $rescale = 1./255$ ,  $brightness\_range = [0.7, 1.0]$ ,  $shear\_range = 0.2$ ,  $zoom\_range = 0.2$ ). Data Augmentation is an effective way to increase training data.
3. *Image analysis using CNN*: the model was built using five convolutional blocks with  $kernel\_size = 3 \times 3$ , activation function is “Relu”,  $pool\_size = 2 \times 2$  and uses “Dropout” to reduce overfitting with  $rate = 0.2$ . The last layer uses the activation function is “Sigmoid” because of binary classification. The optimization chosen was “Adam”. Using Callbacks including ModelCheckpoint to save a copy of the best model, and EarlyStopping to stop model training when the difference between “training” and “validation error” starts to increase, instead of decreasing. The detail of this step is presented in Fig. 4.
4. *Compile and build the model*: implementing the model training process.
5. *Test model on test data*: use the “confusion matrix” and get the results for precision, recall, accuracy.



**Fig. 3** Process flow of the proposed method



**Fig. 4** Process flow of image analysis using convolutional neural network

6. *Check and evaluate results:* in this step, we check and evaluate the results of the proposed method.

To evaluate the results of the model, we use evaluation scales such as accuracy, precision and recall. A confusion matrix is used to present the values obtained after the model testing process, with these values, we can calculate the effectiveness of the training model. The components of the confusion matrix are described in detail in Fig. 5.

We apply the diagnosis of pneumonia to explain these four indicators. In the diagnosis of pneumonia, we have two layers: the pneumonia layer and the normal layer. The grade of pneumonia was marked as positive, and the normal layer was marked as negative. We define the parameters for the evaluation results such as True Positive (TP), False Negative (FN), True Negative (TN) and False Positive (FP).

- TP: Actual image is pneumonia, and the predicted model is pneumonia.
- FN: Actual image is pneumonia, but it is predicted to be normal.
- TN: Actual image is normal, and it is predicted to be normal.
- FP: Actual image is normal, but it is predicted to be pneumonia.

Accuracy is the ratio of the number of pneumonia predictor data that is correctly divided by the total number of pneumonia data in the test dataset. The accuracy

**Fig. 5** Confusion matrix [20]

		Predicted Values	
		Normal (0)	Pneumonia (1)
Actual Values	Normal (0)	TN	FP
	Pneumonia (1)	FN	TP

method is only suitable for general assessment. If we need to get the number of details of each type of prediction is true or false, this method has not provided enough information. For example, a model with 95% accuracy can be either very good or bad depending on actual data classification requirements [20, 21]. The value of accuracy is calculated as Eq. (3):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

The precision is the ratio of the number of cases that the pneumonia image prediction model was true to the actual dataset divided by the total number of cases the model predicted pneumonia images. The value of precision is calculated as Eq. (4)

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

The recall is the ratio of the number of cases that correctly predicted pneumonia divided by the total number of pneumonia image cases in the real dataset. The value of recall is calculated as Eq. (5):

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

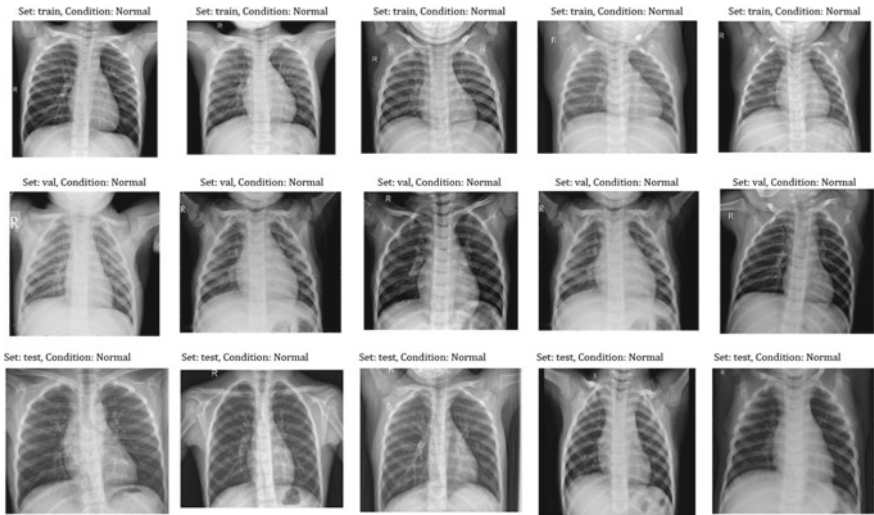
In the diagnosis of pneumonia, if the recall rate is very high, the efficiency of the model will be very good, because the model can detect most cases of pneumonia.

## 4 Experiments and Results

In this section, we experiment and evaluate the results of the proposed method. We compare the results of the proposed method with the results of the other methods, such as Ayan method [10] and Kadam method [12].

### 4.1 The Dataset for Training and Testing

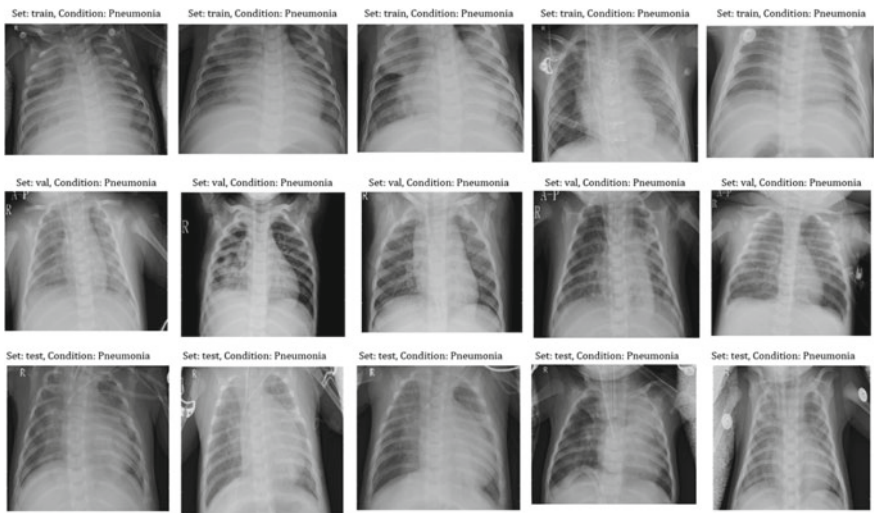
We use X-ray dataset provided by Kaggle [22]. This dataset contains 5863 chest X-ray (JPEG) images with different resolutions. The dataset is structured into three parts, such as: training, testing and validation. Each part containing subfolders for each category is pneumonia or normal. X-ray images have been manually classified by doctors and divided into two categories. The images in the dataset have different resolutions, such as from  $407 \times 178$  to  $2916 \times 2583$  pixel. There are 1583 normal cases, 4273 cases of pneumonia images in the dataset. In our study, all images were



**Fig. 6** X-ray image of the lungs normal case

scaled down to  $150 \times 150$  pixels to maintain the same aspect ratio between all images. Some images from the dataset are shown in Fig. 6 including fifteen normal images, and Fig. 7 including fifteen images of pneumonia. The number of medical images for training, testing and validation present in Table 1.

Now, we test the proposed method with the above dataset and compare with the other methods, such as Ayan method [10] and Kadam method [12].



**Fig. 7** X-ray image of lungs pneumonia case



**Table 1** Illustrates the number of images in three parts

	Train	Validation	Test	Total
Normal	1342	9	234	1583
Pneumonia	3876	9	390	4273

**Table 2** The configuration of CNN networks

Configuration	Values
Training data	All
Input size	150 × 150
Batch size	32
Epoch	32
Step per epoch	163
Kernel size	3 × 3
Activation function	Relu
Activation function of the last layer	Sigmoid
Optimizer	Adam
Loss function to use for classification	binary_crossentropy

## 4.2 Experiments

Our experiments developed on Google Colab, including resources such as 13 GB of RAM, 2 CPUs of 2.2 GHz, GPU and TPU 12 GB. Dataset X-ray images stored on Google Drive cloud. Methods to build training models and classify diseases through six steps as in Sect. 3. Table 2 present the configuration information for the parameters in the model.

Table 3 describes the layers in CNN, including one input layer, five convolution blocks, fully connected layer and output layer.

Information on model test results is presented in “confusion matrix” in Fig. 8.

Now, we compared the results of the proposed method with the results of the other methods. The results comparisons are presented in Table 4.

According to the Table 4 presented the above, our experimental results are better than the experimental results of Ayan method [10] and Kadam method [12] on the same dataset.

With the results of Ayan method [10], the results of the proposed method are better than this method on all three indicators: accuracy, precision, and recall. With the results of Kadam method [12], the value of recall of Kadam method [12] is 99.27% while the value of the proposed method is 98.72%. However, both the precision and accuracy values of the proposed method are 91.23% and 93.27%, which are greater than the precision and accuracy values of the Kadam method are 90.88% and 92.90%.

**Table 3** Convolutional neural network architecture

Layer (type)	Output shape	Parameters
InputLayer	(None, 150, 150, 3)	0
Conv2d_1_1	(None, 150, 150, 16)	448
Conv2d_1_2	(None, 150, 150, 16)	2320
Max_pooling2d_1	(None, 75, 75, 16)	0
Separable_conv2d_2_1	(None, 75, 75, 32)	688
Separable_conv2d_2_2	(None, 75, 75, 32)	1344
Batch_normalization_2	(None, 75, 75, 32)	128
Max_pooling2d_2	(None, 37, 37, 32)	0
Separable_conv2d_3_1	(None, 37, 37, 64)	2400
Separable_conv2d_3_2	(None, 37, 37, 64)	4736
Batch_normalization_3	(None, 37, 37, 64)	256
Max_pooling2d_3	(None, 18, 18, 64)	0
Separable_conv2d_4_1	(None, 18, 18, 128)	8896
Separable_conv2d_4_2	(None, 18, 18, 128)	17,664
Batch_normalization_4	(None, 18, 18, 128)	512
Max_pooling2d_4	(None, 9, 9, 128)	0
Dropout_4	(None, 9, 9, 128)	0
Separable_conv2d_5_1	(None, 9, 9, 256)	34,176
Separable_conv2d_5_2	(None, 9, 9, 256)	68,096
Batch_normalization_5	(None, 9, 9, 256)	1024
Max_pooling2d_5	(None, 4, 4, 256)	0
Dropout_5	(None, 4, 4, 256)	0
Flatten_6	(None, 4096)	0
Dense_7	(None, 1024)	4,195,328
Dropout_8	(None, 1024)	0
Dense_9	(None, 512)	524,800
Dropout_10	(None, 512)	0
Dense_11	(None, 128)	65,664
Dropout_12	(None, 128)	0
Dense_13 (Output)	(None, 1)	129

**Fig. 8** Confusion matrix result

		Predicted Values	
		Normal	Pneumonia
Actual Values	Normal	197	37
	Pneumonia	5	385

**Table 4** The table compares the results of researches on the chest X-ray image

Method	Dataset chest X-ray	Accuracy (%)	Precision (%)	Recall (%)
Ayan method [10]	Kaggle	87.00	91.00	89.00
Kadam method [12]	Kaggle	92.90	90.88	99.27
Proposed method	Kaggle	93.27	91.23	98.72

The reason which the results of the proposed method has better results than the other method because the classification results of pneumonia are higher in the evaluation methods.

## 5 Conclusions

Lung diseases image classification is hard work and challenge. In this paper, we developed an automated method of checking chest X-ray images with high accuracy, using the dataset Chest X-ray images. These images have been pre-classified by radiologists and divided into two types: pneumonia and normal images. We use deep learning methods, build models starting from steps such as pre-processing, image acquisition, image layering, setting appropriate parameters for each layer such as image resolution, number of training times, optimal algorithms, threshold changes, and state transitions during training to get the best and most appropriate model. The test results are highly effective image classification model. The results of the proposed method have high precision and better than those recently published on the same dataset of chest X-ray images. In the future work, we will extend the experiment to update the number of layers, the time of training or testing on the other pre-train models such as: InceptionNet, ResNet, InceptionResNet, MobileNet and CheXNet. Those tasks aim to reduce the training time and increase the level accuracy of the model compared to the above results.

**Acknowledgements** This research is funded by University of Cuu Long.

**Conflicts of Interest** The authors have no conflict of interest to declare.

## References

1. Neuman MI et al (2012) Variability in the interpretation of chest radiographs for the diagnosis of pneumonia in children. *J Hosp Med* 7(4):294–298
2. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*, pp 1097–1105
3. Selvathi D, Aarthi Poornila A (2017) Deep learning techniques for breast cancer detection using medical image analysis. In: *Biologically rationalized computing techniques for image processing applications*. Springer, Berlin, pp 159–186

4. AliKadampur M, Al Riyae S (2020) Skin cancer detection: applying a deep learning based model driven architecture in the cloud for classifying dermal cell images. In: Informatics in medicine unlocked, vol 18. Elsevier, Amsterdam
5. Le QV, Ranzato M, Monga R, Devin M, Chen K, Corrado GS, Dean J, Ng AY (2012) Building high-level features using large scale unsupervised learning. In: Proceedings of the 29th international conference on international conference on machine learning, Edinburgh, Scotland, UK, pp 507–514
6. Katamreddy S, Doody P, Riordan D (2018) Visual udder detection with deep neural networks. In: 12th international conference on sensing technology. IEEE, Ireland, pp 166–171
7. Lujan-Garcia JE, Yanez-Marquez C, Villuendas-Rey Y, Camacho-Nieto O (2020) A transfer learning method for pneumonia classification and visualization, artificial intelligence for medical image analysis. *Appl Sci* 10(8)
8. Sirazitdinov I, Kholiavchenko M, Mustafaev T, Yixuan Y, Kuleev R, Ibragimov B (2019) Deep neural network ensemble for pneumonia localization from a large-scale chest x-ray database. In: Computers and electrical engineering, vol 78. Elsevier, Amsterdam, pp 388–399
9. Sirish Kaushik V, Nayyar A, Kataria G, Jain R (2019) Pneumonia detection using convolutional neural networks. In: Proceedings of first international conference on computing, communications, and cyber-security, vol 121. Springer, Berlin, pp 471–483
10. Ayan E, Unver HM (2019) Diagnosis of pneumonia from chest X-ray images using deep learning. In: Scientific meeting on electrical-electronics & biomedical engineering and computer science. IEEE, Turkey
11. Ponnada VT, Naga Srinivasu SV (2019) Edge AI system for pneumonia and lung cancer detection. *Int J Innov Technol Exploring Eng* 8(9)
12. Kadam K, Ahirrao S, Kaur H, Phansalkar S, Pawar A (2019) Deep learning approach for prediction of pneumonia. *Int J Sci Technol Res* 8(10):86–89
13. Ji S, Xu W, Yang M, Yu K (2013) 3D convolutional neural networks for human action recognition. *IEEE Trans Pattern Anal Mach Intell* 35(1):221–231
14. Adiyansjah, Sgunawan AA, Suhartono D (2019) Music recommender system based on genre using convolutional recurrent neural networks. *Procedia Comput Sci* 157:99–109
15. Ramaswamy S, DeClerck N (2018) Customer perception analysis using deep learning and NLP. *Procedia Comput Sci* 140:170–178
16. Xu Z, Zhang J, Wang J, Xu Z (2020) Prediction research of financial time series based on deep learning. *Soft Comput* 24:8295–8312
17. CNN lectures. <https://cs231n.github.io/convolutional-networks>. CS231n, Stanford. Last accessed 2020/04/28
18. Classification Convolutional Neural Network (CNN). <https://www.thelearningmachine.ai/cnn>. Last accessed 2020/04/28
19. Ayan E, Unver HM (2018) Data augmentation importance for classification of skin lesions via deep learning. In: 2018 electric electronics, computer science, biomedical engineering's meeting. IEEE, Turkey, pp 1–4
20. Powers DMW (2011) Evaluation: from precision, recall and F-measure to ROC, informedness, markedness & correlation. *J Mach Learn Technol* 2(1):37–63
21. Confusion Matrix in Machine Learning. <https://www.geeksforgeeks.org/confusion-matrix-machine-learning>. Last accessed 2020/04/12
22. Kaggle Chest X-Ray Images. <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>. Last accessed 2020/04/24