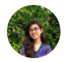
COVID-19 Detection from Chest X-Ray Images Using CNN Models and Deep Learning

Chapter · December 2023


Nafisha Binte Moin
East West University (Bangladesh)


Shamima Sultana
East West University (Bangladesh)


Abdullah Al Munem
East West University (Bangladesh)


Ahmed Wasif Reza
East West University (Bangladesh)

COVID-19 Detection from Chest X-Ray Images Using CNN Models and Deep Learning

Nafisha Binte Moin[1], Shamima Sultana[1], Abdullah Al Munem[1], Omar Tawhid Imam[2], Ahmed Wasif Reza[1], and Mohammad Shamsul Arefin[3,4](✉)

[1] Department of Computer Science and Engineering, East West University, Dhaka, Bangladesh
wasif@ewubd.edu

[2] Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh

[3] Department of Computer Science and Engineering, Daffodil International University, Dhaka 1341, Bangladesh
sarefin@cuet.ac.bd

[4] Department of Computer Science and Engineering, Chittagong University of Engineering and Technology, Chattogram 4349, Bangladesh

Abstract. The dangerous disease and ailment known as COVID-19 has caused a global pandemic and immeasurable damage to people all around the globe. Chest X-Rays are being used to identify COVID-19 in current times due to their low cost and efficiency. In this paper, we developed Convolutional Neural Network models to detect COVID-19 from Chest X-Ray images. CNN models from the Keras library such as VGG16, VGG19, Xception, ResNet101, ResNet152, Inception, InceptionResNet, MobileNetV2, DenseNet201, NASNetLarge, and EfficientNetB3 have been used to perform experimentations. The CNN models used ImageNet as its pre-trained weights for transfer learning. Additionally, a multi-layered self-designed model has been implemented as well to see the performance. A comparative analysis has been completed in order to find the best-performing CNN model for COVID-19 detection in Chest X-Ray images. From the experiments, we found that the proposed CNN gave the best results. Additionally, it has been observed that MobileNetV2, Inception, ResNet101, and VGG16 give the highest accuracy over 99%, while the lowest accuracy is found by EfficientNetB3 at only around 50%. The self-designed multi-layered model gives a training accuracy of 97.22% and a validation accuracy of 96.42%. A significant increase in accuracy and excellent performance has been seen from the CNN models and the proposed framework.

Keywords: Convolutional neural network · COVID-19 · Chest X-ray

1 Introduction

In current times, Coronavirus and "COVID-19" is a well-known and feared terms all across the globe. Currently, it is known as the most dangerous and widespread disease all around the world causing the deaths of millions and harm to all sectors and part of

P. Vasant et al. (Eds.): ICO 2023, LNNS 729, pp. 1779–1791, 2023.
https://doi.org/10.1007/978-3-031-36246-0_171

regular life. In general definition, Coronavirus disease (shortly COVID-19) is an airborne and infectious viral disease caused by a virus known as the SARS-CoV-2 virus which mainly affects the lungs of the human respiratory system [1].

The symptoms that are more commonly observed for this disease include fever, cough, loss of taste and smell, and tiredness. There may be pains and difficulty breathing, shortness of breath, and chest pain as well. There are other additional symptoms of this disease as well. This dangerous disease has damaged the medical and financial stability of almost all countries in the world and has caused a global pandemic.

Over the last two years, many methods and treatments have been discovered and researched to treat and cure this deadly disease. Along with treatment and cure, many steps have also been taken to ensure the prevention of this deadly and epidemic disease. One method for the identification and detection of this viral disease is using X-Ray images of the Chest or Lungs to detect the virus. This process can speed up the identification process compared to time-consuming testing in clinics [2].

Over the last few years, the field of artificial intelligence has rapidly advanced in the use of image classification. In the case of medical science, a popular image classification Deep Learning model known as the Convolutional Neural Networks (or CNNs) is widely used. CNNs can be used for the detection of diseases in different organs of human beings and even plants [3]. CNN is an effective tool in the field of medical science in tasks such as image classification, segmentation, and localization with its performance outperforming humans for diseases related to the brain, breast, lungs, and other organs [4].

2 Related Work

COVID-19 has become a widely known and feared term all across the globe since the year 2020. In order to tackle the issues rising due to COVID-19 many previous works have been done in recent years to work on fast prediction and detection of the disease. In most earlier related works, the majority of authors worked on improving the performance of the CNNs for the diagnosis of the disease.

Panwar et al. [5] used an open-source dataset to conduct experimentations on the proposed model 'nCOVnet' with transfer learning. The designed model used VGG16 as its base model and ImageNet as the pre-trained weights. Training accuracy gives 97% accuracy with 98.68% confidence.

Jahid et al. [6] performed experimentations on COVID-19 X-Rays, Pneumonia (another lung disease) X-Rays, and unaffected Chest X-Rays using the models three CNNs. The dataset was from Kaggle, where the images were resized with augmentations. The models' performances were measured by calculating the precision, recall, and other scores of the models which were between the ranges of 0.98–1.

Basu et al. [7], it can be seen that CNNs have been used for the identification of COVID-19 images collected from four image databases. Experimentations on the CNN models such as AlexNet (82.98% accuracy), VGGNet (90.13% accuracy), and Resnet (85.98% accuracy) were performed.

Ismael et al. [8] work on fine-tuning along with end-to-end training of CNNs for disease detection. ResNet and VGG were the two CNN models which were used where feature extraction was performed by using different kernels of SVM. The accuracies of the models range from 85.26 to 92.63%.

Hussain et al. [9] proposed a model (with 22 layers) using X-Rays and CTs of Normal, COVID-19, and Pneumonia as input. CoroDet model gives high accuracy for both training and validation along with high confidence scores.

Nayak et al. [10] perform experiments to test the performance of CNNs for the identification of diseased lungs. The CNNs used namely, AlexNet, VGG16, GoogleNet, MobileNetV2, SqeezeNet, ResNet34, ResNet50, and InceptionV3 has been used to perform the experiments. The paper shows the model ResNet34 performs better than the rest of the models with an accuracy of 98.33%.

Heidari et al. [11] showed that work has been done to improve CNN predictions. Image preprocessing techniques such as histogram equalization algorithms and bilateral low-pass filters are used. The dataset images are used to form pseudo-colored images. The study yields a high confidence interval and high sensitivity.

Minaee et al. [12] used transfer learning techniques. Using Chest radiology images from open datasets experiments were performed on the CNN models namely ResNet (18 and 50), as well as SqueezeNet, and also DenseNet-121. The models received a sensitivity of 98% and a specificity rate of 90%.

Mangal et al. [13] introduced a COVID detection CNN model called CovidAID. The paper works on the use of X-Rays for further testing of RT-PCR. The model achieves an accuracy of around 90.5% after it has been tested on a publically available dataset.

Alazab et al. [14] created datasets that have been used to create the COVID detection model. The models yield an F1 score of around 95–99%. In addition, multiple deep learning methods like the prophet algorithm, ARIMA, and LSTM were implemented to carry out predictions.

Wang et al. [15] used deep learning methods. The model was trained on a dataset created from 13,975 images (collected and compiled from 13,870 patients). Projection and expansion designs have been implemented. In this paper, CNN models VGG and ResNet achieve an accuracy of 83% and 90.6% respectively, while COVID-Net achieved 93.3% accuracy.

Zhang et al. [16] use deep learning by using the X-Ray image dataset from a GitHub repository. The developed models show a sensitivity of 96% for COVID-19-positive cases and a sensitivity of 70.65% for COVID-19-negative cases.

Tabik et al. [17] use a dataset known as the COVIDGR dataset which implemented the COVID-SDNet model. Here, the dataset has been created by collaborating with a hospital. The model achieves results of 97.72% (severe), 86.90% (moderate), and 61.80% (mild) accuracy in different severity levels.

Abbas et al. [18] performed COVID-19 diagnosis using the DeTrac CNN model. The deep learning technique transfer learning has also been used. The model can deal with irregularities and achieved a very high accuracy of around 93.1%.

Alghamdi et al. [19] performed a survey about using Deep Learning and CNNs for disease detection from Chest X-Rays. The study highlights the necessity of diverse datasets which should be publicly available. The common CNN models that were popular for experimentation among researchers are ResNet, DenseNet, GoogleNet/Inception, and VGGNet. The work in [21–25] focuses different techniques those deployed mainly image analysis techniques.

3 System Architecture and Design

Table1 highlights the proposed framework of the study, showing layers and types of outputs shape and the total count of parameters used. The model contains a total of 20 layers where there are 6 are Convolutional, 4 of them are Max Pooling, and 6 Dropout. Then Flatten Layers and Dense Layer have also been used for the final predictions of the model.

Table 1. Model architecture of proposed model

Layer	Output shape	Parameters
Convolution	(None, 329, 329, 32)	896
Convolution	(None, 327, 327, 64)	18,496
Max pooling	(None, 163, 163, 64)	0
Dropout	(None, 163, 163, 64)	0
Convolution	(None, 161, 161, 64)	36,928
Max pooling	(None, 80, 80, 64)	0
Dropout	(None, 80, 80, 64)	0
Convolution	(None, 78, 78, 128)	73,856
Max pooling	(None, 39, 39, 128)	0
Dropout	(None, 39, 39, 128)	0
Convolution	(None, 37, 37, 128)	147,584
Max pooling	(None, 18, 18, 128)	0
Dropout	(None, 18, 18, 128)	0
Convolution	(None, 16, 16, 128)	147,584
Max pooling	(None, 8, 8, 128)	0
Dropout	(None, 8, 8, 128)	0
Flatten	(None, 8192)	0
Dense	(None, 64)	524,352
Dropout	(None, 64)	0
Dense	(None, 2)	130
Total parameters		949,826
Trainable parameters		949,826
Non-trainable parameters		0

Table 1 shows the layers used for the model. In addition to the self-designed model, the dataset has experimented on multiple built-in Keras CNN models such as VGG16, VGG19, Xception, ResNet101V2, ResNet152V2, InceptionV3, InceptionResNetV2, MobileNetV2, DenseNet201, NASNetLarge, and EfficientNetB3. The models used the Batch Normalization layers (renorm = True) and Global Average Pooling2D layers. Then, layers such as Dense Layers and Dropouts were used. The activations Relu and Softmax were used for the dense layers.

3.1 Dataset Description

The dataset for disease detection dataset was obtained and collected from the data science platform Kaggle [20]. The dataset contains three directories where the images from the training and validation dataset were used for the study to train the models. The dataset contains two class labels, one for diseased lung X-rays and another for healthy X-ray images.

The training and validation files have an equal distribution of images. In this dataset, a total of 348 images have been used as the inputs for the proposed method and the experimental CNN models.

Figure 1 displays some of the sample images for both healthy and diseased X-Ray images from the training directory of the dataset.

Table 2 shows detailed information about the distribution of images throughout the dataset for both Covid and Normal class labels in the case of both training and validation image datasets.

In total, 348 image samples were collected and used for analysis and experimentation. All the images are in png format in this open-sourced dataset. Total Covid images are 174 and healthy Chest X-Ray images are 174. A completely balanced distribution of images has been observed for the collected dataset as there is an equal number of images for both Covid and Normal samples.

3.2 Data Preprocessing

In the proposed method for this study, a total of 348 images of both class labels have been used. For preprocessing the dataset and in order to make it suitable for training, the images have been resized where the input images have a size of 331 $\times$ 331 pixels. The images were rescaled.

Data augmentation using the ImageDataGenerator function of Keras has been performed on the training and validation images. Horizontal and vertical flips have been applied after rescaling and the selected color mode was RBG. The training and validation split of the dataset was already created and separated into respective directories of the dataset.

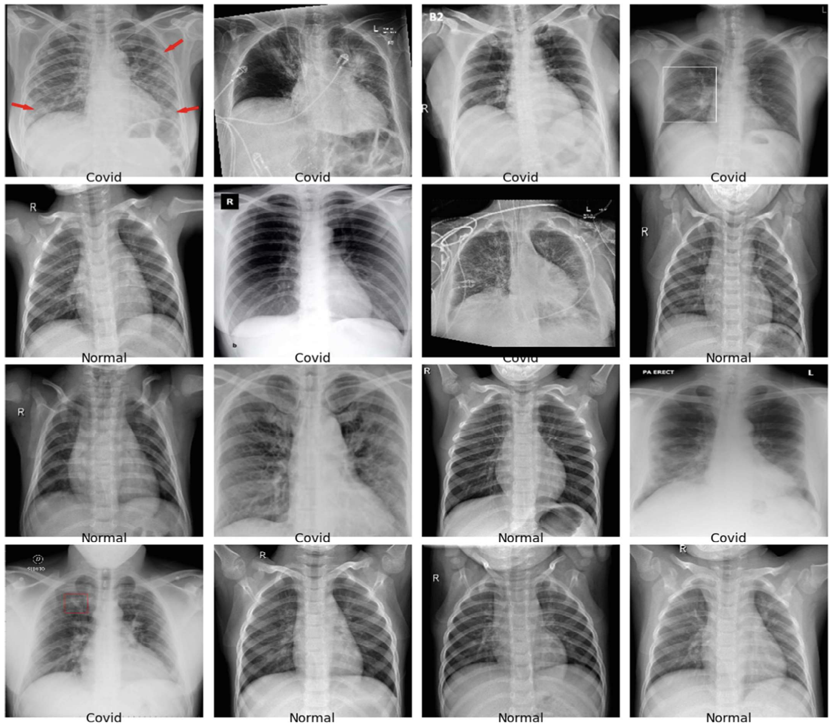


Fig. 1. Sample input images data set (train)

Table 2. Image distribution of dataset

Train/validation set	Class labels	Total images
Train dataset	Covid	144
	Normal	144
Validation dataset	Covid	30
	Normal	30
Total		348

4 Implementation and Experimental Result

4.1 Experimental Setup

The experimentations were performed using the online platform Google Colaboratory. Google Colab contains the required latest version of Python along with additional dependencies and libraries such as Keras and Tensorflow. Keras has been used to create the CNN models and perform all the experimentations.

The total number of epochs is 25, with a batch size equal to 8. Adam was the optimizer and cross-entropy (categorical) was the loss function. The chosen metric for evaluating the model performance for each epoch was the accuracy measure. The callback functions (early-stopping) with patience and restoring best weights have also been implemented into the model. The models were evaluated on all the images in training, validation, and testing.

4.2 Performance Evaluation

In order to accurately and comprehensively evaluate the efficiency of the system and all the experimented CNN models, the accuracy has been calculated using the following equations All the calculated values were obtained automatically from the Keras model training period. In addition to the accuracy, the calculation of the loss, precision, recall, and f1-score have also been performed for model evaluation.

$$Accuracy = \frac{No.\ of\ corrected\ correspondence}{No.\ of\ correspondence} \times 100\% \tag{1}$$

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{2}$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{3}$$

$$F1 - score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \tag{4}$$

The metrics and evaluation scores with the equations show how well the model is performing along with how good the model is for real-world experimentations and implementation.

Table 3 shows that the accuracy of the proposed framework surpasses many previous works and is around 96.77%. The proposed framework gives a very low loss where the average loss is approximately 0.10429. Additionally, a high value of the other metrics can be observed which exceeds more than 93%. The obtained f1-score for the proposed system is 95%. In the case of precision, Covid-19 positive cases received a percentage of 94% and negative cases had 97%. While for recall, Covid-19 positive case is 97% and the negative case is 93%.

Figure 2 shows the accuracies while Fig. 3 shows the losses over 25 epochs in both train and validation sets. The proposed framework shows an exponential increase in accuracy and a slow and smooth decrease in loss over the epochs. The graphs plotting the accuracy and loss for the epochs show an exponential curve. The model generalizes and learns from the data without errors. Along with the proposed model results, the results for the experimental built-in Keras models are briefly discussed in the following tables highlighting the accuracy, loss, etc.

In Table 4, almost all the CNN models give a very high accuracy except Efficent-NetB3. Out of all the models, VGG16, ResNet101V2, InceptionV3, and MobileNetV2 give the highest accuracy of 99%. Almost all the models here give higher accuracy than the proposed framework. VGG19, ResNet152V2, InceptionResNetV2, and NASNet-Large have the second-highest values of 97%. The most underperforming CNN model is the EfficientNetB3 having less than 50% accuracy. There was a decrease in accuracy

Table 3. Performance analysis for proposed system

Train accuracy	97.22%
Train loss	0.09587
Validation accuracy	96.42%
Validation loss	0.09375
Validation evaluation accuracy	96.67%
Validation evaluation loss	0.12327
Precision (COVID)	0.94
Precision (normal)	0.97
Recall (COVID)	0.97
Recall (normal)	0.93
F1-score (COVID)	0.95
F1-score (normal)	0.95

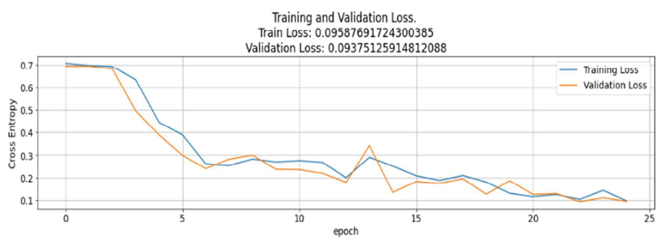


Fig. 2. Training and validation loss for proposed framework

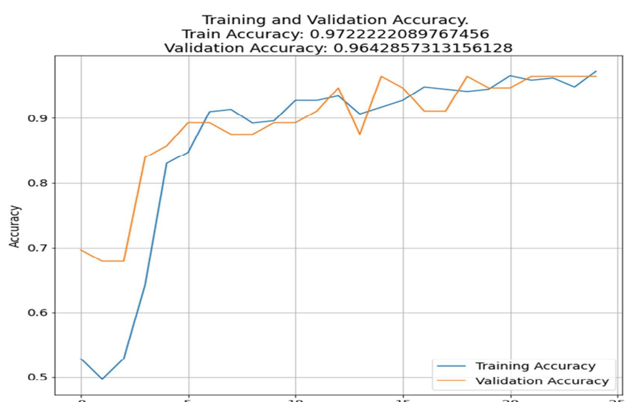


Fig. 3. Training and validation accuracy for proposed framework

Table 4. Performance analysis for built-in Keras CNN models

Models	Train accuracy (%)	Validation accuracy (%)	Validation evaluation accuracy (%)
VGG16	99.65	100.00	98.33
VGG19	93.40	91.07	91.67
Xception	98.95	96.42	95.00
ResNet101V2	99.30	98.21	96.67
ResNet152V2	97.22	94.64	95.00
InceptionV3	98.95	96.42	96.67
InceptionResNetV2	97.22	98.21	96.67
MobileNetV2	99.65	98.21	98.33
DenseNet201	98.26	96.42	98.33
NASNetLarge	96.87	96.42	96.67
EfficientNetB3	46.52	48.21	50.00

percentages for the models when using validation data. The validation accuracies are similar to the training accuracies.

Table 5 shows the loss calculated for all the models during training, validation, and evaluation. The lower the value of the model loss, the more efficiently the model is performing and the less prone to show errors. Low loss determines how the model generalizes per epoch and improves performance and learning. The table shows that VGG16 and MobileNetV2 have the lowest loss values. Therefore, they are performing better compared to the other models. Xception, ResNet101V2, and InceptionV3 also have a comparatively lower loss, while ResNet152V2, InceptionResNetV2, DenseNet201, and NASNetLarge have comparatively higher loss values. EfficientNetB3 is the worst-performing model with a loss higher than 0.69.

Table 6 shows the values obtained for all the mentioned metrics in the case of all the models. The values were obtained from the validation data. It can be seen that the scores for all the models and the 20-layered framework are comparatively higher than the required benchmark scores for this evaluation. Xception, ResNet152V2, and InceptionResNetV2 give comparatively higher scores where the values range from 94 to 100%. EfficientNetB3 gives the worst results with an f1-score of 67%.

Tables 3 through 6 highlight the experimental results for all the CNN models and self-designed novel proposed framework along with a comparison of which models performed the best and which models performed the worst for the identification of the disease from the dataset images.

Table 5. Loss for built-in Keras CNN models

Models	Train loss	Validation loss	Validation evaluation loss
VGG16	0.15574	0.14944	0.17977
VGG19	0.27254	0.31001	0.30134
Xception	0.19411	0.23464	0.26204
ResNet101V2	0.18148	0.19645	0.22280
ResNet152V2	0.20594	0.25528	0.24951
InceptionV3	0.19447	0.23364	0.23008
InceptionResNetV2	0.23756	0.21918	0.24458
MobileNetV2	0.17348	0.19489	0.19274
DenseNet201	0.20210	0.23204	0.20017
NASNetLarge	0.23647	0.24278	0.23910
EfficientNetB3	0.69470	0.69357	0.69323

Table 6. Precision, recall, and F1-score for built-in models

Models	Precision (C)	Precision (N)	Recall (C)	Recall (N)	F1-score (C)	F1-score (N)
VGG16	0.94	1.00	1.00	0.93	0.97	0.97
VGG19	1.00	0.94	0.93	1.00	0.97	0.97
Xception	0.97	1.00	1.00	0.97	0.98	0.98
ResNet101V2	0.97	1.00	1.00	0.97	0.98	0.98
ResNet152V2	0.88	1.00	1.00	0.87	0.94	0.93
InceptionV3	0.86	1.00	1.00	0.83	0.92	0.91
InceptionResNetV2	0.97	1.00	1.00	0.97	0.98	0.98
MobileNetV2	0.97	1.00	1.00	0.97	0.98	0.98
DenseNet201	0.94	1.00	1.00	0.93	0.97	0.97
NASNetLarge	0.97	1.00	1.00	0.97	0.98	0.98
EfficientNetB3	0.00	0.50	0.00	1.00	0.00	0.67

4.3 Comparison with Other Existing Frameworks

Table 7 given below highlights the accuracies in other existing frameworks in contrast to the proposed framework and the built-in models with the previous works for both self-designed models and well-established CNN models.

From the comparison, it can be said that most of the experimental models are showing high accuracies compared to the previous related works. VGG16, MobileNetV2, and ResNet101 show better performances compared to the previous works. The proposed

Table 7. Comparison with previous works

Paper No.	Previous works results (%)	Previous works models	Experimented models performance (%)
[5]	97.00	Self-designed	96.77
[7]	90.13	VGG16	99.32
[7]	85.98	ResNet101	98.06
[8]	85.26	VGG16	99.32
[8]	87.37	ResNet101	98.06
[8]	89.47	VGG19	92.04
[9]	99.10	Self-designed	96.77
[10]	95.83	VGG16	99.32
[10]	95.83	MobileNetV2	98.73
[10]	92.50	InceptionV3	97.34
[15]	83.00	VGG19	92.04
[15]	93.30	Self-designed	96.77

framework's accuracy does not exceed the previously designed frameworks but there is a very slight difference between the accuracies of the models.

5 Conclusion

In this study, 11 popular Convolutional Neural Networks such as VGG16, VGG19, Xception, ResNet101V2, ResNet152V2, InceptionV3, InceptionResNetV2, MobileNetV2, DenseNet201, NASNetLarge, EfficientNetB3 were used. In addition to the eleven CNN models, a self-designed framework has been. All the models except EfficicentNetB3 give high accuracy and low loss. Most of the models outperform the previous works. The framework designed for this work gives an accuracy of 96.77% with a loss of 0.10429.

In conclusion, the proposed framework and the built-in Keras models, especially VGG16, MobileNetV2, and ResNet101V2 give excellent performance in the damaged and diseased lungs from Chest X-Ray images for Covid. All those models can be implemented and tested on different datasets to compare and improve performances. From the eleven experimental and built-in Keras models used for this study, VGG16, MobileNetV2, and ResNet101V2 outperform the rest of the models. The proposed framework carries the lowest loss value out of all the models.

Limitations to this study are fewer data usage for training. More data along with additional class labels for Pneumonia and Tuberculosis will further improve the prediction range and working range. Additionally, various techniques for transforming images can be applied to improve performance. These proposed techniques may also improve the precision, recall, and f1 scores. Additionally, kit tests can also be used together with this method for the accuracy of results. Future work for this study would include creating a web interface or a mobile application where the models will be used to conduct predictions in real time and as an end to end models.

References

1. WHO: WHO coronavirus disease (COVID-19). https://www.who.int/health-topics/coronavirus. Last accessed 21 May 2022
2. Rahaman, M.M., et al.: Identification of COVID-19 samples from chest X-ray images using deep learning: a comparison of transfer learning approaches. J. X-Ray. Sci. Technol. **28**, 821–839 (2020). https://doi.org/10.3233/XST-200715
3. Moin, N.B., Islam, N., Sultana, S., Chhoa, L.A., Ruhul Kabir Howlader, S.M., Ripon, S.H.: Disease detection of Bangladeshi crops using image processing and deep learning—a comparative analysis. In: 2022 2nd International Conference on Intelligent Technologies. CONIT 2022, pp. 1–8 (2022). https://doi.org/10.1109/CONIT55038.2022.9847715
4. Sarvamangala, D.R., Kulkarni, R.V.: Convolutional neural networks in medical image understanding: a survey. Evol. Intel. **15**(1), 1–22 (2021). https://doi.org/10.1007/s12065-020-00540-3
5. Panwar, H., Gupta, P.K., Siddiqui, M.K., Morales-Menendez, R., Singh, V.: Application of deep learning for fast detection of COVID-19 in X-rays using nCOVnet. Chaos Solitons Fractals **138**, 109944 (2020). https://doi.org/10.1016/j.chaos.2020.109944
6. Hasan Jahid, M., Alom Shahin, M., Ali Shikhar, M.: Deep learning based detection and segmentation of COVID-19 pneumonia on chest X-ray image. In: 2021 International Conference on Information and Communication Technology for Sustainable Development. ICICT4SD 2021—Proceedings, pp. 210–214 (2021). https://doi.org/10.1109/ICICT4SD50815.2021.9396878
7. Basu, S., Mitra, S., Saha, N.: Deep learning for screening COVID-19 using chest X-ray images. In: 2020 IEEE Symposium Series on Computational Intelligence. SSCI 2020, pp. 2521–2527 (2020). https://doi.org/10.1109/SSCI47803.2020.9308571
8. Ismael, A.M., Şengür, A.: Deep learning approaches for COVID-19 detection based on chest X-ray images. Expert Syst. Appl. **164** (2021). https://doi.org/10.1016/j.eswa.2020.114054
9. Hussain, E., Hasan, M., Rahman, M.A., Lee, I., Tamanna, T., Parvez, M.Z.: CoroDet: a deep learning based classification for COVID-19 detection using chest X-ray images. Chaos Solitons Fractals **142**, 110495 (2021). https://doi.org/10.1016/j.chaos.2020.110495
10. Nayak, S.R., Nayak, D.R., Sinha, U., Arora, V., Pachori, R.B.: Application of deep learning techniques for detection of COVID-19 cases using chest X-ray images: a comprehensive study. Biomed. Signal Process. Control. **64**, 102365 (2021). https://doi.org/10.1016/j.bspc.2020.102365
11. Heidari, M., Mirniaharikandehei, S., Khuzani, A.Z., Danala, G., Qiu, Y., Zheng, B.: Improving the performance of CNN to predict the likelihood of COVID-19 using chest X-ray images with preprocessing algorithms. Int. J. Med. Inform. **144**, 104284 (2020). https://doi.org/10.1016/j.ijmedinf.2020.104284
12. Minaee, S., Kafieh, R., Sonka, M., Yazdani, S., Jamalipour Soufi, G.: Deep-COVID: predicting COVID-19 from chest X-ray images using deep transfer learning. Med. Image Anal. **65** (2020). https://doi.org/10.1016/j.media.2020.101794
13. Mangal, A., Kalia, S., Rajgopal, H., Rangarajan, K., Namboodiri, V., Banerjee, S., Arora, C.: CovidAID: COVID-19 Detection Using Chest X-Ray, pp. 1–10 (2020)
14. Alazab, M., Awajan, A., Mesleh, A., Abraham, A., Jatana, V., Alhyari, S.: COVID-19 prediction and detection using deep learning. Int. J. Comput. Inf. Syst. Ind. Manag. Appl. **12**, 168–181 (2020)

15. Wang, L., Lin, Z.Q., Wong, A.: COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images. Sci. Rep. **10**, 1–12 (2020). https://doi.org/10.1038/s41598-020-76550-z
16. Zhang, J., et al.: Viral pneumonia screening on chest X-rays using confidence-aware anomaly detection. IEEE Trans. Med. Imaging **40**, 879–890 (2021). https://doi.org/10.1109/TMI.2020.3040950
17. Tabik, S., et al.: COVIDGR dataset and COVID-SDNet methodology for predicting COVID-19 based on chest X-ray images. IEEE J. Biomed. Heal. Inform. **24**, 3595–3605 (2020). https://doi.org/10.1109/JBHI.2020.3037127
18. Abbas, A., Abdelsamea, M.M., Gaber, M.M.: Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network. Appl. Intell. **51**(2), 854–864 (2020). https://doi.org/10.1007/s10489-020-01829-7
19. Alghamdi, H.S., Amoudi, G., Elhag, S., Saeedi, K., Nasser, J.: Deep learning approaches for detecting COVID-19 from chest X-ray images: a survey. IEEE Access. **9**, 20235–20254 (2021). https://doi.org/10.1109/ACCESS.2021.3054484
20. Fenta, F.: Chest X-ray for covid-19 detection. https://www.kaggle.com/datasets/fusicfenta/chest-xray-for-covid19-detection. Last accessed 15 May 2022
21. Saha, R., Debi, T., Arefin, M.S.: Developing a framework for vehicle detection, tracking and classification in traffic video surveillance. In: Vasant, P., Zelinka, I., Weber, GW. (eds.) Intelligent Computing and Optimization. ICO 2020. Advances in Intelligent Systems and Computing, vol. 1324. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-68154-8_31
22. Fatema, K., Ahmed, M.R., Arefin, M.S.: Developing a system for automatic detection of books. In: Chen, J.I.Z., Tavares, J.M.R.S., Iliyasu, A.M., Du, K.L. (eds.) Second International Conference on Image Processing and Capsule Networks. ICIPCN 2021. Lecture Notes in Networks and Systems, vol. 300. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-84760-9_27
23. Rahman, M., Laskar, M., Asif, S., Imam, O.T., Reza, A.W., Arefin, M.S.: Flower recognition using VGG16. In: Chen, J.I.Z., Tavares, J.M.R.S., Shi, F. (eds.) Third International Conference on Image Processing and Capsule Networks. ICIPCN 2022. Lecture Notes in Networks and Systems, vol. 514. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-12413-6_59
24. Yeasmin, S., Afrin, N., Saif, K., Imam, O.T., Reza, A.W., Arefin, M.S.: Image classification for identifying social gathering types. In: Vasant, P., Weber, G.W., Marmolejo-Saucedo, J.A., Munapo, E., Thomas, J.J. (eds.) Intelligent Computing & Optimization. ICO 2022. Lecture Notes in Networks and Systems, vol. 569. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-19958-5_10
25. Ahmed, F., et al.: Developing a classification CNN model to classify different types of fish. In: Vasant, P., Weber, G.W., Marmolejo-Saucedo, J.A., Munapo, E., Thomas, J.J. (eds.) Intelligent Computing & Optimization. ICO 2022. Lecture Notes in Networks and Systems, vol. 569. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-19958-5_50