



Unsupervised Convolutional Filter Learning for COVID-19 Classification

Sakthi Ganesh Mahalingam^{1*}, Saichandra Pandraju²

¹ Vellore Institute of Technology, Vellore 632014, India

² QIS College of Engineering and Technology, Ongole 523272, India

Corresponding Author Email: saichandrapandraju@gmail.com

<https://doi.org/10.18280/ria.350509>

Received: 29 August 2021

Accepted: 13 October 2021

Keywords:

COVID-19, CAE, CNN, LSTM, Chest X-Ray

ABSTRACT

The outbreak of the SARS CoV-2, referred to as COVID-19, was initially reported in 2019 and has swiftly spread around the world. The identification of COVID-19 cases is one of the key factors to inhibit the spread of the virus. While there are multiple ways to diagnose COVID-19, these techniques are often expensive, time-consuming, or not readily available. Detection of COVID-19 using a radiological examination of Chest X-Rays provides a more viable, rapid, and efficient solution as it is easily available in most countries. The paper outlines a method that employs an unsupervised convolutional filter learning using Convolutional Autoencoder (CAE) followed by applying it to COVID-19 classification as a downstream task. This shows that the proposed technique provides state-of-the-art results with an average accuracy of 99.7%, AUC of 99.7%, specificity of 99.8%, sensitivity of 99.6%, and F1-score of 99.6%. We release the data and code for this work to aid further research.

1. INTRODUCTION

COVID-19 is one of the most highly infectious diseases in the last decade. As per the World Health Organization (WHO) report, as of July 22, 2021, over 192 million cases have been confirmed, with over 4.1 million deaths worldwide. This alarming rate of infection spread calls for critical and efficient diagnosis to curb the spread of the virus. Currently, the majority of the cases are being diagnosed using techniques such as RT-PCR, LFT, and Antibody Testing, which often produce highly accurate results. However, these techniques are generally time-consuming and expensive, posing a significant issue for developing and under-developed countries.

Chest X-Rays offer an alternate screening method for detecting COVID-19, where they are analyzed to look for visible evidence associated with SARS CoV-2 viral infection. The recent studies of the chest radiographic images show that viruses belonging to this family demonstrate substantial manifestation in radiographic images [1]. Thus, the categorization with the help of Chest X-rays could be a potentially cost-effective and accurate solution.

In our paper, we propose Convolutional Autoencoding as a pre-training technique for X-ray images. This model will learn to compress an image using an encoder, and again, decompress it to the original form using a decoder. Due to the unsupervised nature of this approach, the potential cost of labeling the images is alleviated. The resulting encoder will be capable of extracting complex features from an image, which will then be used as a pre-trained network, along with transfer learning to classify images. We also apply Gradual Unfreezing for the fine-tuning process as different layers in a network learn different features. For example, the lower layers of the CNN learn basic features such as edges and lines, while the higher layers learn more complex features such as textures and patterns. So, instead of refining all layers at once, which risks

catastrophic interference, Gradual Unfreezing [2] suggests slowly un-freezing the model starting from the last layer as it contains the least general knowledge.

2. RELATED WORK

Recently, several AI systems based on deep learning have been implemented with promising results to detect COVID-19 using chest radiography images. Rahimzadeh and Attar [3] proposed a concatenated neural network of Xception and ResNet50V2 by using 180 COVID chest X-rays which obtained an accuracy of 99.56% and recall of 80.56%. A Generative Adversarial Network (GAN) approach was introduced by Loey et al. [4] using GoogleNet, AlexNet, and ResNet18 in low data settings with 69, 79, 79, 79 images for Covid, bacterial pneumonia, viral pneumonia, and normal cases respectively. This approach achieved test accuracy of 80.6% with GoogleNet, 85.2% with AlexNet, 99.9% with GoogleNet for 4-class, 3-class, and 2-class scenarios respectively. Ucar and Korkmaz [5] used SqueezeNet with Bayesian optimization with 76 COVID X-ray images and achieved 98.3% accuracy. Apostolopoulos and Mpesiana [6] implemented transfer learning with CNN using VGG19, Inception, MobileNet, Xception, and Inception-ResNetV2 and selected VGG19 as the final model with accuracy, sensitivity, specificity of 93.48%, 98.75%, and 92.85% respectively. Bandyopadhyay and Dutta [7] introduced an LSTM-GRU framework for confirming positive, negative, death and release cases of COVID with an accuracy of 87%, 67.8%, 62%, and 40.5% respectively.

Khan et al. [8] proposed CoroNet, a Deep CNN model, using Xception architecture with 284, 330, 327, 310 images for COVID, bacterial pneumonia, viral pneumonia, and normal cases respectively and achieved an accuracy of 85.9%,

a precision of 97% and a recall of 100%. Horry et al. [9] presented a COVID classification system using Xception, VGG, ResNet, and Inception and got an accuracy of 80%. Hemdan et al. [10] proposed COVIDX-Net, a framework of Deep Learning Classifiers, utilizing 7 Deep CNN architectures - VGG19, ResNetV2, InceptionV3, Inception-ResNetV2, MobileNetV2, DenseNet201, and Xception and achieved scores of 90% and 83% for accuracy and precision respectively. Singh et al. [11] performed COVID classification with 4 classifiers - Deep CNN, Extreme Learning Machine (ELM), Online sequential ELM, and Bagging Ensemble with Support Vector Machine (SVM). Bagging Ensemble with SVM is the best performing model with 95.7% accuracy and 95.8% precision. Ahuja et al. [12] proposed a 3 phase-detection model consisting of data augmentation phase, COVID detection phase, and abnormality localization phase to improve detection accuracy and achieved a test dataset accuracy of 99.4%. Fong et al. [13] conducted a case study that used composite Monte-Carlo (CMC) and fuzzy rule induction addressing the data limitation for forecasting. Islam et al. [14] proposed Deep CNN and CNN-LSTM models and achieved an accuracy of 99.4%, 99.9% AUC, 99.2% specificity, 99.3% sensitivity, and F1-score of 98.9% with the CNN-LSTM model.

3. RESEARCH METHOD

In this section, we present the proposed methodology for the identification of COVID-19 from chest X-Rays using an Unsupervised pre-training approach. Our proposed technique relies on a variant of Autoencoding called Convolutional Auto-encoder (CAE). An autoencoder has two main parts: an encoder that maps the input into the code and a decoder that maps the code to reconstruct the input. An autoencoder is a specific type of artificial neural network mainly employed to handle unsupervised machine learning tasks. In particular, an autoencoder is a feedforward neural network that is trained to predict the input itself. Thus, the system can minimize the reconstruction error by ensuring the hidden units capture the most appropriate features of the data.

Autoencoding is a data compression algorithm in which both compression and decompression are data-specific and automatically learned from samples instead of human engineering. This type of automatic or unsupervised learning is paramount to domains having less supervised data but huge unsupervised data. In our approach, we use Convolutional

Neural Networks (CNNs or convnets) for compression and decompression as they have a proven record for images, and this variant is Convolutional Autoencoder (CAE). Figure 1 shows the block diagram of the Convolutional Auto-Encoder consisting of an Encoder and Decoder model.

The encoder tries to compress the input image by extracting important features such that the decoder can recreate the original image with minimal loss. This acts as a pre-training objective that allows the encoder to extract important features.

After training this encoder-decoder model on the training data, the encoder model is saved, and the decoder model is discarded. This encoder can then be used as a data preparation technique to perform feature extraction on the raw data, which in turn can be used to train a different machine learning model for downstream tasks.

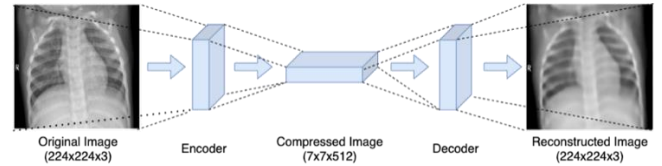


Figure 1. Block diagram of convolutional auto-encoder model

The proposed technique achieves this in a two-step process:

- (1) Unsupervised Pre-training of Convolutional Auto-Encoder.
- (2) Transfer Learning with Encoder for downstream classification task – Detection of Chest X-Rays affected with COVID-19.

3.1 Unsupervised pre-training of convolutional auto-encoder

Dataset: We collected 18617 chest X-ray images from various sources – covid19-radiography-database [15], covid19-chest-xray-image-dataset [16], chest-xray-pneumonia [17], covid-chestxray-dataset [18], Figure 1-COVID-chestxray-dataset [19] and Actualmed-COVID-chestxray- dataset [20].

Modeling:

The Encoder (E) takes the input image ‘x’, and compresses it into lower-dimensional features ‘s’ known as ‘bottleneck’:

$$s = E(x) \tag{1}$$

Table 1. Architecture of encoder block

| Layer (type) | Input Shape | Kernel # | Kernel Size | Output Shape | Param # |
|-----------------------------|-----------------|----------|-------------|-----------------|---------|
| Conv2D | 224 x 224 x 3 | 64 | 3 x 3 | 224 x 224 x 64 | 1792 |
| Conv2D | 224 x 224 x 64 | 64 | 3 x 3 | 224 x 224 x 64 | 36928 |
| MaxPooling2D | 224 x 224 x 64 | - | 2 x 2 | 112 x 112 x 64 | 0 |
| Conv2D | 112 x 112 x 64 | 128 | 3 x 3 | 112 x 112 x 128 | 73856 |
| MaxPooling2D | 112 x 112 x 128 | - | 2 x 2 | 56 x 56 x 128 | 0 |
| Conv2D | 56 x 56 x 128 | 256 | 3 x 3 | 56 x 56 x 256 | 295168 |
| MaxPooling2D | 56 x 56 x 256 | - | 2 x 2 | 28 x 28 x 256 | 0 |
| Conv2D | 28 x 28 x 256 | 512 | 3 x 3 | 28 x 28 x 512 | 1180160 |
| MaxPooling2D | 28 x 28 x 512 | - | 2 x 2 | 14 x 14 x 512 | 0 |
| Conv2D | 14 x 14 x 512 | 512 | 3 x 3 | 14 x 14 x 512 | 2359808 |
| MaxPooling2D | 14 x 14 x 512 | - | 2 x 2 | 7 x 7 x 512 | 0 |
| Total params: 3,947,712 | | | | | |
| Trainable params: 3,947,712 | | | | | |
| Non-trainable params: 0 | | | | | |

Table 2. Architecture of decoder block

| Layer (type) | Input Shape | Kernel # | Kernel Size | Output Shape | Param # |
|-----------------------------|----------------|----------|-------------|----------------|---------|
| Conv2DTranspose | 7 x 7 x 512 | 512 | 3 x 3 | 14 x 14 x 512 | 2359808 |
| Conv2DTranspose | 14 x 14 x 512 | 256 | 3 x 3 | 28 x 28 x 256 | 1179904 |
| Conv2DTranspose | 28 x 28 x 256 | 128 | 3 x 3 | 56 x 56 x 128 | 295040 |
| Conv2DTranspose | 56 x 56 x 128 | 64 | 3 x 3 | 112 x 112 x 64 | 73792 |
| Conv2DTranspose | 112 x 112 x 64 | 3 | 3 x 3 | 224 x 224 x 3 | 1731 |
| Total params: 3,910,275 | | | | | |
| Trainable params: 3,910,275 | | | | | |
| Non-trainable params: 0 | | | | | |

Even though most X-ray images are of single channel, it is possible for the images to have markings or coloration to highlight specific portions. Training the encoder layers with 3-channel images instead of single-channel images allows us to apply the pretrained encoder model for various downstream X-ray applications. Hence, we resize the input X-ray images to (224x224x3) and rescale all the pixel values in the range 0-1 for normalization. This ensures that each input has a similar data distribution and also improves computational efficiency. This preprocessed image is then fed to the Encoder (E). We design the Encoder as a 6-layer 2D Convolutional Network with Rectified Linear Unit (ReLU) as activation function, kernel size of (3x3), and Max-Pooling with a pool size of (2x2). This 6-layer Convolutional Encoder Network outputs (7x7x512) extracted features as shown in Table 1.

The Decoder (D) accepts the lower dimensional extracted features as inputs and reconstructs the original image with shape (224x224x3). Our Decoder as shown in Table 2 is a 5-layer Transposed 2D Convolutional Network with a stride of 2, ReLU activation function, and a kernel size of (3x3). This kernel size allows the model to learn complex features with less computation.

If we denote reconstructed image as ‘y’, then,

$$y = D(s) \quad (2)$$

From Eq. (1) and Eq. (2), the output image ‘y’ can be denoted as:

$$y = D(E(x)) \quad (3)$$

The reconstructed image ‘y’ is then compared with input ‘x’ and loss is calculated which is used to update the weights of all the layers. We used ‘Mean Squared Error’ (MSE) as the loss function. The MSE represents the cumulative squared error between the compressed and the original image, and lower the value of MSE, the lower the error. The Mean Squared Error can be calculated using the following equation:

$$MSE = \frac{\sum_{M,N} [I_1(m,n) - I_2(m,n)]^2}{M * N} \quad (4)$$

where, I_1 and I_2 are the input and output images with dimensions (m, n) respectively. ‘Adam’ is used as the optimizer to update weights with an initial learning rate of 0.01.

3.2 Transfer learning with encoder for downstream classification task – Detection of Chest X-Rays affected with COVID-19

Dataset: We collected 3709 COVID-19 chest X-rays from various sources - covid19-radiography-database [15], covid19-chest-xray-image-dataset [16], Figure1-COVID-chestxray-dataset [19] and Actualmed-COVID-chestxray-dataset [20]. Then we collected 3700 images for both normal and pneumonia cases from ‘covid19-radiography-database [15] and chest-xray-pneumonia [17] respectively. Similar to the data preparation step during pre-training, we resized all images to (224,224,3) and rescaled each pixel to be in 0-1 range as a preprocessing step.

Table 3. Architecture of proposed classification network

| Layer (type) | Input Shape | Kernel # | Kernel Size | Output Shape | Param # |
|---------------------------------|-----------------|----------|-------------|-----------------|---------|
| Conv2D | 224 x 224 x 3 | 64 | 3 x 3 | 224 x 224 x 64 | 1792 |
| Conv2D | 224 x 224 x 64 | 64 | 3 x 3 | 224 x 224 x 64 | 36928 |
| MaxPooling2D | 224 x 224 x 64 | - | 2 x 2 | 112 x 112 x 64 | 0 |
| Conv2D | 112 x 112 x 64 | 128 | 3 x 3 | 112 x 112 x 128 | 73856 |
| MaxPooling2D | 112 x 112 x 128 | - | 2 x 2 | 56 x 56 x 128 | 0 |
| Conv2D | 56 x 56 x 128 | 256 | 3 x 3 | 56 x 56 x 256 | 295168 |
| MaxPooling2D | 56 x 56 x 256 | - | 2 x 2 | 28 x 28 x 256 | 0 |
| Conv2D | 28 x 28 x 256 | 512 | 3 x 3 | 28 x 28 x 512 | 1180160 |
| MaxPooling2D | 28 x 28 x 512 | - | 2 x 2 | 14 x 14 x 512 | 0 |
| Conv2D | 14 x 14 x 512 | 512 | 3 x 3 | 14 x 14 x 512 | 2359808 |
| MaxPooling2D | 14 x 14 x 512 | - | 2 x 2 | 7 x 7 x 512 | 0 |
| Reshape | 7 x 7 x 512 | - | - | 49 x 512 | 0 |
| LSTM | 49 x 512 | - | - | 49 x 512 | 2099200 |
| Flatten | 49 x 512 | - | - | 25088 | 0 |
| Dense | 25088 | - | - | 64 | 1605696 |
| Dense (output) | 64 | - | - | 3 | 195 |
| Total params: 7,652,803 | | | | | |
| Trainable params: 3,705,091 | | | | | |
| Non-trainable params: 3,947,712 | | | | | |

Modelling: In this section, we explain the process of transfer learning with the pre-trained encoder for COVID-19 classification. The encoder model from the previous step is attached to a classification head for downstream classification. Here, we chose LSTM and Dense Layers that result in a network that is very efficient for classification tasks. Figure 2 shows the block diagram of the classification network with output shapes of each layer. In Figure 2, the input image is a test image fed to the pre-trained encoder obtained from the trained CAE model. This encoder extracts the important features from the image and outputs a compressed image of size (7x7x512). Table 3 shows the detailed architecture along with the total trainable parameters.

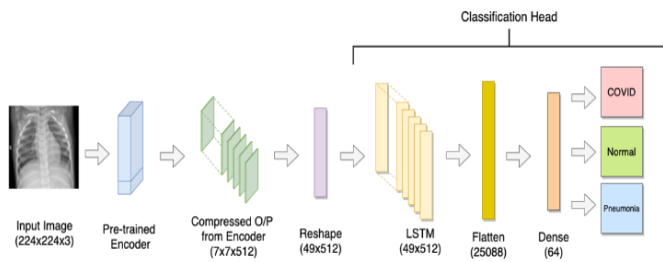


Figure 2. Block diagram of proposed convolutional network

This classification model is fine-tuned by applying the ‘Gradual Unfreezing’ technique. As the weights of the classification head (LSTM & Dense) are randomly initialized, we first freeze all the encoder layers, so that the pre-trained weights are not affected. Once the classification head is trained, we unfreeze the encoder layers and train the entire model. This technique helps the model to learn layer-level features effectively as the lower layers learn simple representations such as edges and curves while the higher layers learn complex representations.

4. RESULTS

In this experiment, the Chest X-Ray dataset was split into 73-13.5-13.5 sets for training, validation, and testing respectively. The proposed architecture is a combination of 6 Convolutional layers, an LSTM, and 2 Dense layers as shown in Table 3, trained using a single NVIDIA Tesla T4 with 16GB memory. We used ‘Adam’ optimizer with a learning rate of 0.001, categorical cross-entropy loss, early stopping with patience of 5, and batch size of 16. The total fine-tuning process took only 25 minutes and 18 epochs to converge because of the efficient pre-training method.

We experimented with various architectures for the classification head and Table 4 shows the performance comparison of these models. Although the proposed model is the best performing, it is worth acknowledging the competitiveness of these architectures because of the robustness of the pre-trained encoder.

Figure 3 shows the confusion matrix of the proposed method. Figure 4 shows the graphical representation of the performance metrics. Most of the existing systems rely on models that were pretrained on image datasets not specific to X-ray images. Hence, these models rely entirely on the fine-

tuning phase to learn about the features present in X-rays. Whereas the proposed CAE architecture is pretrained on a large corpus of X-ray images, allowing the model to extract intricate features useful for a wide range of X-ray related downstream tasks. Table 5 corroborates the effectiveness of the proposed system by comparing it with contemporary architectures.

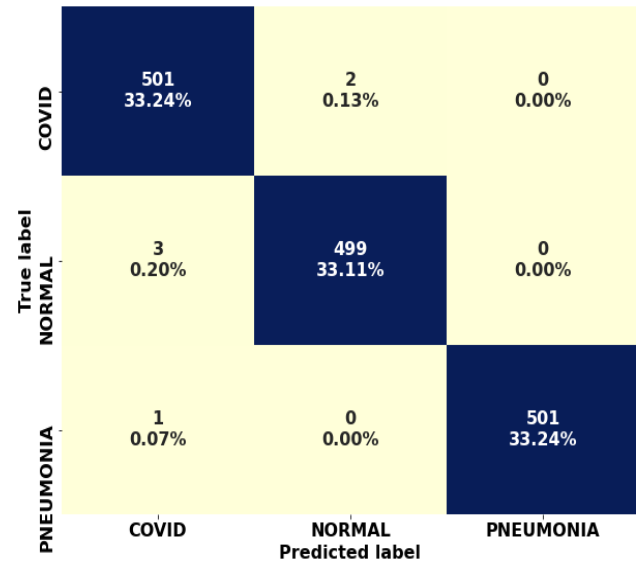


Figure 3. Confusion matrix of proposed model

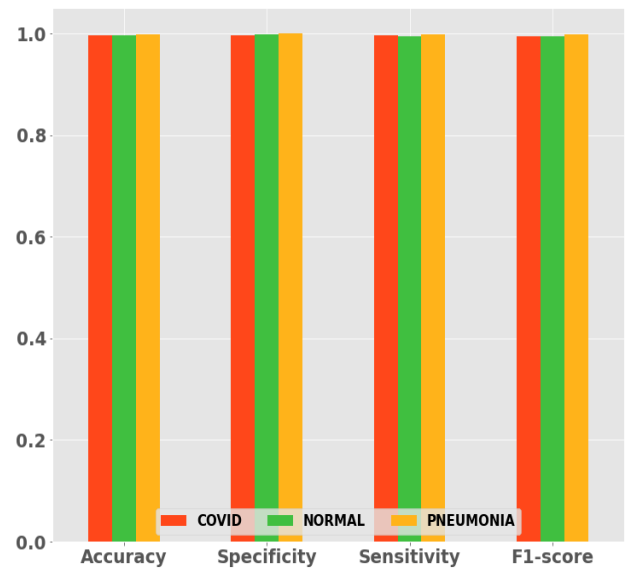


Figure 4. Representation of performance metrics

Table 4. Performance comparison of classification heads

| Classification Head | Accuracy (%) | Precision (%) | Recall (%) | F1-score (%) | AUC (%) |
|-----------------------------|--------------|---------------|------------|--------------|---------|
| 3 LSTM + 2 Dense | 97.5 | 96.3 | 96.3 | 96.3 | 97.2 |
| 1 LSTM + 3 Dense | 97.7 | 96.6 | 96.6 | 96.6 | 97.4 |
| 5 Dense | 98.4 | 97.5 | 97.6 | 97.55 | 98.2 |
| 1 LSTM + 2 Dense (Proposed) | 99.7 | 99.6 | 99.6 | 99.6 | 99.7 |

Table 5. Comparison of proposed system with existing systems

| Author | Architecture | Training Samples (COVID-19) | Testing Samples (COVID-19) | Accuracy (%) | Accuracy (COVID-19) (%) |
|------------------------|-----------------------|--------------------------------|-------------------------------|--------------|----------------------------|
| Rahimzadeh et al. | ResNet50V2 + Xception | 180 | 31 | 91.4 | 99.6 |
| Loey et al. | GoogleNet | 69 | 9 | 80.6 | 100.0 |
| Ucar et al. | COVIDiagnosis-Net | 76 | 10 | 98.3 | 100.0 |
| Khan et al. | CoroNet | 284 | 29 | 89.5 | 96.6 |
| Hemdan et al.. | COVIDX-Net | 25 | 5 | 90.0 | - |
| Islam et al. | CNN-LSTM | 1220 | 305 | 99.4 | 99.2 |
| Proposed System | CAE | 2703 | 503 | 99.73 | 99.6 |

5. CONCLUSION

In this paper, we suggest an unsupervised pre-training technique for diagnosing COVID-19 from chest X-rays. The proposed technique utilizes a combination of Convolutional Neural Network (CNN) which can be used as a pre-trained network to extract features for any chest X-ray tasks and an LSTM layer and Dense layer as classification head. According to the experiment results, the proposed method achieved an accuracy of 99.73%, a precision of 99.6%, a sensitivity of 99.6%, and an AUC of 99.7% with an F1-score of 99.6%. We hope that the proposed system would be able to help patients and reduce the workload of the medical diagnosis of COVID-19. Finally, the performance of our proposed system was not compared with radiologists and that would be part of a future study.

REFERENCES

- [1] Das, A.K., Ghosh, S., Thunder, S., Dutta, R., Agarwal, S., Chakrabarti, A. (2021). Automatic COVID-19 detection from X-ray images using ensemble learning with convolutional neural network. *Pattern Analysis and Applications*, 24: 1111-1124. <https://doi.org/10.1007/s10044-021-00970-4>
- [2] Howard, J., Ruder, S. (2018). Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pp. 328-339. <https://doi.org/10.18653/v1/P18-1031>
- [3] Rahimzadeh, M., Attar, A. (2020). A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2. *Informatics in Medicine Unlocked*, 19: 100360. <https://doi.org/10.1016/j.imu.2020.100360>
- [4] Loey, M., Smarandache, F., M Khalifa, N.E. (2020). Within the lack of chest COVID-19 X-ray dataset: A novel detection model based on GAN and deep transfer learning. *Symmetry*, 12(4): 651. <https://doi.org/10.3390/sym12040651>
- [5] Ucar, F., Korkmaz, D. (2020). COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease 2019 (COVID-19) from X-ray images. *Medical Hypotheses*, 140: 109761. <https://doi.org/10.1016/j.mehy.2020.109761>
- [6] Apostolopoulos, I.D., Mpesiana, T.A. (2020). COVID-19: Automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. *Physical and Engineering Sciences in Medicine*, 43(2): 635-640. <https://doi.org/10.1007/s13246-020-00865-4>
- [7] Bandyopadhyay, S.K., Dutta, S. (2020). Machine learning approach for confirmation of covid-19 cases: Positive, negative, death and release. *MedRxiv*. <https://doi.org/10.1101/2020.03.25.20043505>
- [8] Khan, A.I., Shah, J.L., Bhat, M.M. (2020). CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images. *Computer Methods and Programs in Biomedicine*, 196: 105581. <https://doi.org/10.1016/j.cmpb.2020.105581>
- [9] Horry, M.J., Chakraborty, S., Paul, M., Ulhaq, A., Pradhan, B., Saha, M., Shukla, N. (2020). X-ray image based COVID-19 detection using pre-trained deep learning models. <https://doi.org/10.31224/osf.io/wx89s>
- [10] Hemdan, E.E.D., Shouman, M.A., Karar, M.E. (2020). Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images. *arXiv preprint arXiv: 2003.11055*. <http://arxiv.org/abs/2003.11055>
- [11] Singh, M., Bansal, S., Ahuja, S., Dubey, R.K., Panigrahi, B.K., Dey, N. (2021). Transfer learning-based ensemble support vector machine model for automated COVID-19 detection using lung computerized tomography scan data. *Medical & Biological Engineering & Computing*, 59(4): 825-839. <https://doi.org/10.1007/s11517-020-02299-2>
- [12] Ahuja, S., Panigrahi, B.K., Dey, N., Rajinikanth, V., Gandhi, T.K. (2021). Deep transfer learning-based automated detection of COVID-19 from lung CT scan slices. *Applied Intelligence*, 51(1): 571-585. <https://doi.org/10.1007/s10489-020-01826-w>
- [13] Fong, S.J., Li, G., Dey, N., Crespo, R.G., Herrera-Viedma, E. (2020). Composite Monte Carlo decision making under high uncertainty of novel coronavirus epidemic using hybridized deep learning and fuzzy rule induction. *Applied Soft Computing*, 93: 106282. <https://doi.org/10.1016/j.asoc.2020.106282>
- [14] Islam, M.Z., Islam, M.M., Asraf, A. (2020). A combined deep CNN-LSTM network for the detection of novel coronavirus (COVID-19) using X-ray images. *Informatics in Medicine Unlocked*, 20: 100412. <https://doi.org/10.1016/j.imu.2020.100412>
- [15] <https://www.kaggle.com/tawsifurrahman/covid19-radiography-database>.
- [16] <https://www.kaggle.com/alifrahman/covid19-chest-xray-image-dataset>.
- [17] <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>.
- [18] Cohen, J.P., Morrison, P., Dao, L., Roth, K., Duong, T. Q., Ghassemi, M. (2020). Covid-19 image data collection: Prospective predictions are the future. *arXiv preprint arXiv:2006.11988*. <https://arxiv.org/abs/2003.11597>
- [19] <https://github.com/agchung/figure1-COVID-chestxray-dataset>.
- [20] <https://github.com/agchung/Actualmed-COVID-chestxray-dataset>.