Deep Learning for COVID-19

B. S. Prashanth, M. V. Manoj Kumar, Likewin Thomas, M. A. Ajay Kumar, Dinghao Wu, B. Annappa, Anirudh Hebbar, and Y. V. Srinivasa Murthy

Abstract Ever since the outbreak in Wuhan, China, a variant of Coronavirus named "COVID 19" has taken human lives in millions all around the world. The detection of the infection is quite tedious since it takes 3–14 days for the symptoms to surface in patients. Early detection of the infection and prohibiting it would limit the spread to only to Local Transmission. Deep learning techniques can be used to gain insights on the early detection of infection on the medical image data such as Computed Tomography (CT images), Magnetic resonance Imaging (MRI images), and X-Ray images collected from the infected patients provided by the Medical institution or from the publicly available databases. The same techniques can be applied to do the analysis of infection rates and do predictions for the coming days. A wide range of open-source pre-trained models that are trained for general classification or segmentation is available for the proposed study. Using these models with the concept of transfer learning, obtained resultant models when applied to the medical image datasets would draw much more insights into the COVID-19 detection and prediction process. Innumerable works have been done by researchers all over the world on the publicly available COVID-19 datasets and were successful in deriving

B. S. Prashanth (✉) · M. V. Manoj Kumar · A. Hebbar
Department of Information Science and Engineering, Nitte Meenakshi Institute of Technology, Yelahanka, Bengaluru 560064, Karnataka, India

L. Thomas
Department of Computer Science and Engineering, PES Institute of Technology and Management, Shivamogga 577204, Karnataka, India

M. A. Ajay Kumar · D. Wu
College of Information Sciences and Technology, Pennsylvania State University, University Park, PA 16802, USA
e-mail: duw12@psu.edu

B. Annappa
Department of Computer Science and Engineering, National Institute of Technology Karnataka, Surathkal, Mangalore 575025, Karnataka, India
e-mail: annappa@ieee.org

Y. V. Srinivasa Murthy
Department of Computational Intelligence, School of Computer Science and Engineering, Vellore Institute of Technology (VIT), Vellore 632014, Tamil Nadu, India

J. Nayak et al. (eds.), *Understanding COVID-19: The Role of Computational Intelligence*, Studies in Computational Intelligence 963, https://doi.org/10.1007/978-3-030-74761-9_23

good results. Visualizing the results and presenting the summarized data of prediction in a cleaner, unambiguous way to the doctors would also facilitate the early detection and prevention of COVID-19 Infection.

Keywords Data visualization · Deep learning · CNN · DNN · Transfer learning · Medical imaging · COVID-19

1 Introduction

COVID-19 is a novel strain of coronavirus that has been observed in humans. The COVID-19 had its first outbreak in Wuhan, Hubei province, China [1]. To date, there is no clinically approved medicine that can prevent or cure COVID-19, though the medical practitioners and pharmaceutical agencies are striving hard to develop medicine to cure or to prevent the spread of COVID-19. There are medicines that are being administered to alleviate the progression of fever, cold, and pneumonia-like COVID-19 symptoms. The medical fraternity and the government agencies from various countries have invested huge amounts of revenue and time to invent a medication that can successfully suppress the COVID-19 once and for all. In this appalling time, efforts were made to use Artificial Intelligence and deep learning methodologies to combat the non-medical aspects such as tracking and forecasting outbreaks, detecting the non-compliance of infected patients, containing the outbreak, identification of the hot zones which will help to contain the infection spread. Even in the medical image analysis of the dataset such as CT, MRI, X-Ray images, deep learning, and AI played a pivotal role in the detection, classification, and identification of the infection in a swift manner. In the field of healthcare, the role of AI and deep learning was quintessential and helped the medical institutions and hospitals to provide faster diagnoses in a systematic manner.

Due to the widespread essence of the coronavirus, patients are being admitted to health care in batches. This had pushed the government and medical agencies to their edge to accommodate the higher number of admissions to the medical facility. Setting up an atmosphere where the patient could get quick treatment and in a swift manner is a daunting task. Rapid diagnosis is quintessential and proven to be effective to contain the widespread of the COVID-19 virus. The mortality rate keeps on rising all over the world and it is evident when WHO (World Health Organization) decided to put nCoV as an epidemic disease on February 11, 2020, coiling the term COVID-19 which stands for Coronavirus Disease 2019.

Deep learning techniques were helpful in combating the socio-economic problems that took birth due to the COVID-19 pandemic as well. The tools such as Dashboards are being used by many countries are still helpful for the common people to get information about the precautions to be taken, or the infection rate, fatality rate, etc. These dashboards are internally using various AI and deep learning models for processes such as data gathering or retrieval, assimilation of data, identification of the useful insights from the gathered data from the various sources, and provide meaningful insights based on it in the dashboards. The World Health Organization does use these dashboards extensively to provide all the information about the pandemic to the world [4].

2 Overview of the COVID-19 Pandemic

Coronaviruses are the class of viruses that are known to cause illness such as the common cold, cough, fever, etc. [1, 6]. According to the stats provided by the World Health Organization (WHO) dated Aug 28, there are a total of 24,021,218 positive cases all over the world with the mortality rate is around 821,462 as shown in Fig. 1 [4].

Though on the outset, individuals from all age groups are susceptible to the infection of COVID-19, the probability of the disease being fatal is more around the people aged around 60 and above. The disease had also become very infectious to those who have respiratory disorders and those with chronic medical conditions [5].

The virus spreads through the air medium the most, it spreads when an infected person coughs or sneezes. The droplets sprayed via cough can spread up to 6 feet away. If a healthy person breathes them or swallows them, the virus starts to incubate in that new human body. Some of the infected patients exhibit asymptotic behavior without any symptoms but they still can become a carrier. The virus can also spread through surfaces, so touching a surface with a virus and then followed by touching the mouth, nose or eyes could also lead to the spread of coronavirus [6, 7].

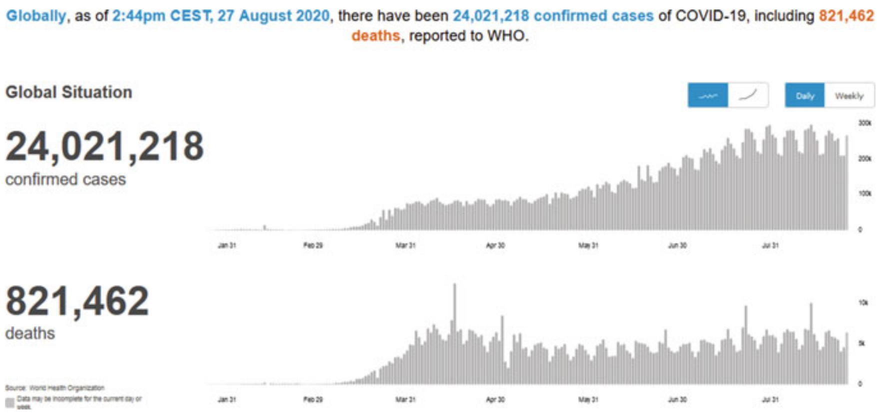


Fig. 1 WHO dashboard summarizing the confirmation cases and deaths due to COVID-19 [4]

3 Introduction to Neural Networks and Deep Learning

To understand deep learning, one needs to understand the difference between rule-based models and data-based models. The complex problems can be solved by imposing a set of mathematically proven rules fed to the machines using which the machines are able to understand and interpret the input data and learn. This is the fundamental idea behind Artificial Intelligence. Contrary to that, machine learning algorithms rely on the input data and they learn the inference based on the quality of the input data. The higher the quality of the data, the higher is the learning. The traditional machine learning algorithms work exceptionally well if there are no well-defined rules. Machine learning algorithms work well for statistical data, or text, numerical data, but they do not yield better results for the other kinds of data such as audio, video, unstructured data, etc. Though perception wise, often Artificial Intelligence and Machine learning are used synonymously, but in fact, Artificial Intelligence is a broader subject that enables and induces smartness to machines and Machine learning is a technique to enable them to be smart. Deep learning is one sub-branch of Machine Learning which comprises Algorithms which is based on mimicking the activities of the human brain.

3.1 Neural Networks

Neural Networks or otherwise known as 'Artificial Neural Networks', are a collection of neurons that identify the complex relationship between input and output. They are loosely inspired by the operations involved in biological neurons of a human brain. The idea of Neural Networks was first started by a psychologist named Frank Rosenblatt, who initially coined the term 'Perceptron' in the year 1958. He observed the similarities between a biological neuron as well as an artificially created neuron. In the next section let us see the architecture of neural networks in detail.

3.1.1 Architecture of Artificial Neural Networks

A typical Neural Network architecture generally constitutes Input Layer, Hidden Layer, Output Layer. Figure 2 depicts a simple Neural Network architecture.

The order of the layers is sequential in nature i.e., the input layer always comes first, then the hidden layers, and finally the output layer. Each layer consists of neurons where complex operations take place which helps in the prediction. The input layer as the name suggests takes in the input of various forms such as numerical values, images, texts, etc. And pass it onto the next layer that is the hidden layer. No mathematical operation takes place in the input layer and a neural network contains only one input layer, but it can have any number of nodes in that layer. The hidden layer is the most important layer of a Neural Network. In this layer, the complex

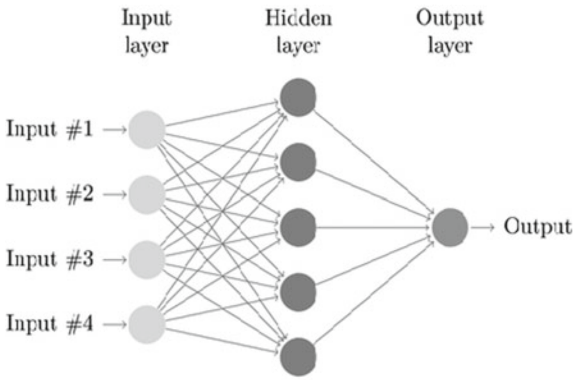


Fig. 2 Simple architecture of Artificial Neural Networks

relationship between the input values and the output value are identified. A prominent step called ‘Feature Extraction’ takes place in these layers where relevant and the most important features from the input values are identified and extracted such that it can be helpful for the network to train on the data and predict accurately for the future, unknown values. Unlike the input layer, the hidden layer can consist of any number of layers along with any number of nodes. The relationship between the input and the output values are generally in a non-linear fashion. To help the Neural Networks to train on the data, special functions known as ‘Activation Function’ are present which introduces the non-linearity in the network. There are many flavors of activation functions such as ‘Sigmoid’, ‘Tanh’, ‘ReLU’, ‘SoftMax’ etc. More details about the activation function will be given in the upcoming section. Finally, in the output layer, the model gives out its prediction based on the extracted features from the data. Just like the input layer, even the neural network contains only one output layer but can have any number of nodes based on the type of problem that is being solved. The process of flow of data starting from the input layer, traversing through the hidden layer, and finally to the output layer is known as the ‘Forward Propagation’. The following section examines the various activation functions which are commonly used in Neural & Deep Neural Networks.

3.1.2 Activation Functions

The activation functions in neural networks introduce non-linearity into the model and aid the network to identify the complex relationship between the input and output. After the computation of input with activation function, the network decides which neurons to fire and in each layer of the hidden layer, thus influencing the final prediction. A few of the prominent activation functions are discussed in this section.

3.1.2.1 Sigmoid Activation Function

This activation is preferred in the output layer when the problem type of binary classification (disease/no disease, fraud/not fraud), since the input from the hidden

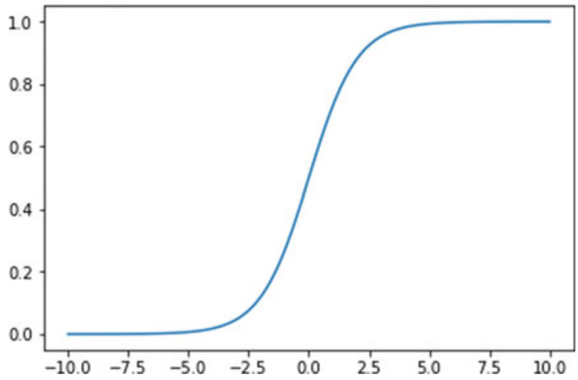


Fig. 3 Sigmoid activation function

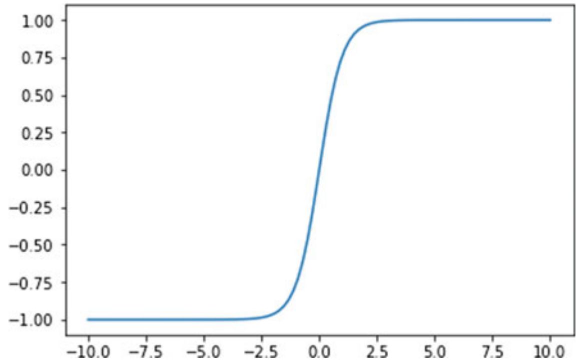


Fig. 4 Tanh activation function

layer, are squashed to the range between 0 and 1 i.e., the probability score for the prediction will lie within these two values. Figure 3 represents the sigmoid activation function.

Equation for sigmoid activation function is given as:

$$\mathrm{y(x)} = 1/e^{(-\mathrm{x})} \tag{1}$$

Where,
y(x) = prediction y is generated for every value of x.

3.1.2.2 Tanh Activation Function

Tanh activation is another popular activation used while training a network. Unlike Sigmoid activation which has a range between (0 to 1), the Tanh activation function has the range between (−1 to1). This centers the data close to zero, thus allowing the layers to learn the features in a better manner. Figure 4 represents the Tanh activation function.

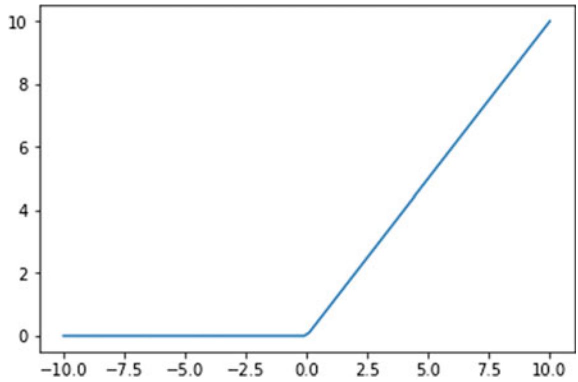


Fig. 5 ReLU activation function

Equation for tanh activation function is given by:

$$y(x) = e^{x} - e^{(-x)}/e^{x} + e^{(-x)} \tag{2}$$

3.1.2.3 ReLU Activation Function

This is perhaps the most popular and frequently used activation function in neural networks and more specifically Convolutional Neural Network. Since ReLU converts all negative pixels in the images to zero and extracts the maximum pixel values, they are helpful in extracting useful features from an image and ignore the noise. ReLU is computationally more feasible compared to sigmoid or tanh and hence faster compared to the other two activation functions. Figure 5 represents the ReLU activation function.

Equation for the ReLU activation function is given by:

$$y(x) = \max(0, x) \tag{3}$$

3.1.2.4 SoftMax Activation Function

SoftMax is one such activation function that is used in the output layer, unlike other activations that can be used both in the hidden layers as well as the output layer. In this activation, the probability scores of each class are given in the output nodes, such that the sum of the probability score is equal to one. Figure 6 represents the SoftMax activation function.

The equation for the SoftMax activation function is given by,

$$P\left(y = j \middle| \theta^{(i)}\right) = e^{\theta(i)} / \sum_{j=0}^{k} e_{k}^{\theta(i)} \tag{4}$$

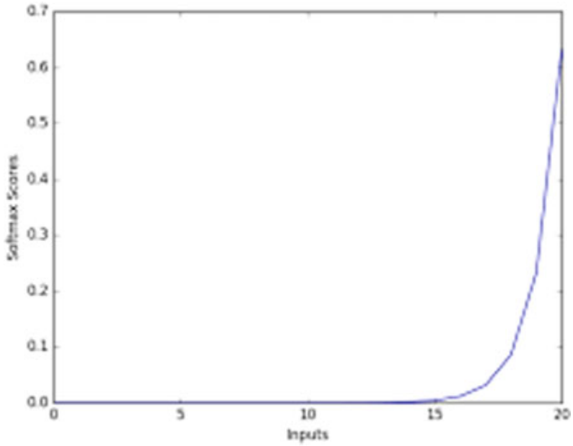


Fig. 6 SoftMax activation function

3.1.3 Advantages and Disadvantages of a Neural Network

Undoubtedly, Artificial Neural Networks (ANN) have offered a robust classification method, this section briefs some of the noteworthy and inherent advantages and disadvantages of ANN.

Advantages of a Neural Network

Neural networks can train by example. They can train with a set of data and can predict the unseen data with ease. Neural networks can generalize. They have the power to predict a new variety of data-based previous trends of data. Neural networks are robust and fault tolerant. They can predict accurately even if the input data has noise or incomplete. Neural networks can understand the complex and non-linear relationship between the input and the output.

Disadvantages of a Neural Network

Neural networks are a black-box model. It means that although accurate predictions can be obtained, it is hard to explain the reason behind that prediction since the relationship between the input and the output is complex and non-linear. Training a neural network is computationally intensive when it is run on a local machine. It mainly depends on the complexity of the network and the amount of data used but does take a lot of time and resources compared to conventional machine learning algorithms. To achieve good results with a neural network a lot more data is required. To develop a custom neural network, a lot of expertise and skill set is required. Thanks to several open-source libraries such as Keras and PyTorch, building a network of certain complexity has been made possible. But these cannot always be relied on and there might arrive a point where the model must be built from scratch. Certain neural network architectures such as Artificial Neural Networks, Convolutional Neural Networks, and Recurrent Neural Networks rely on supervised learning methodology (structured data) to train and predict on any given data and fail to perform well on

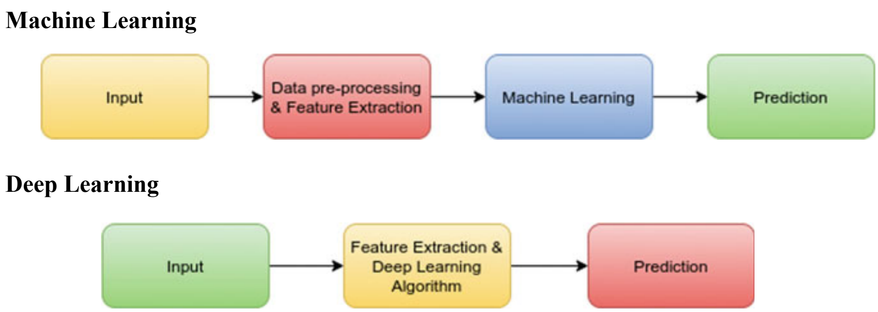


Fig. 7 Notable differences between machine learning and deep learning

those data which do not have labels with them (unsupervised learning). So, to overcome the training of a model via unsupervised methods, there exists another subset of machine learning, known as 'Deep Learning', which can handle both structured as well as unstructured data. This will be discussed in the upcoming section.

3.2 Deep Learning

Deep Learning is a subset of Machine Learning which can be used to train in supervised, semi-supervised, and unsupervised models. This is inspired by Artificial Neural Networks. Deep Learning can extract features from the data in a hierarchical fashion i.e., at first low-level features are extracted, then mid-level features, and finally high-level features are extracted.

3.2.1 What Is Deep Learning?

The main difference between machine learning and deep learning is that in machine learning, the dataset should be pre-processed thoroughly before using it to train the model. In deep learning, the data pre-processing step is not mandatory since the models are robust to noise and missing data, little to no data preprocessing is required. Figure 7 depicts the differences in a clear manner.

In the current scenario, both deep learning and neural networks are so entwined to each other that the terms are used interchangeably, although there are some key differences between the two terminologies which will be discussed in the next section.

3.2.2 Differences Between Deep Learning and Neural Network

Some of the key differences between deep learning and neural networks are. Deep learning is a subset of machine learning which allows the computer systems to improve with experience and data. Neural Networks are inspired by the working mechanism of the nervous system. Deep learning is a type of machine learning method which represents artificial networks. Neural networks are biologically inspired networks that are assigned to perform certain tasks. Deep learning is associated with data transformation and feature extraction, whereas neural networks use neurons to obtain input values, perform operations and provide the output value. Some of the famous architectures in neural networks are Perceptron, Feed-Forward Network, Hopfield Network, Boltzmann machine, etc.

3.2.3 Architectures in Deep Learning

During the last two decades advancement in learning algorithms have given us various deep learning architectures, this helped us to expand the number and type of problems that neural networks could solve. Deep learning is not a stand-alone approach, but it is a range of classification algorithms that one can apply to range of problems.

3.2.3.1 Convolutional Neural Network

CNN is a Deep-Learning algorithm that is used to deal with image data. They are commonly used in image classification, object detection, etc. They extract important features from the data and based on the information obtained they perform prediction. Following are the steps involved in CNN: *Convolution operation.* To achieve this kernel/filter is used over the input image where the filter multiplies its overlapping values by traversing over each pixel of an input image and outputs its sum as a single value until the entire image is traversed. With more added layers it can extract high-level features and understand the data. *Pooling* – This step is similar to Convolution operation i.e.; it helps in reducing the spatial size of the convolved feature. These are also helpful since it extracts the most important features irrespective of rotational and positional variance.

3.2.3.2 Recurrent Neural Network

Recurrent Neural Network is a class of deep learning which specializes in handling sequential data. This feature makes this architecture unique compared to other deep learning architecture since they are not well equipped to handle sequential data. In certain architectures such as feed-forward networks, the prediction is made based on current input, whereas in RNNs, the prediction is based on both current input as well as the information it has learned from the previous input, thus RNN architecture is said to have a short-term memory. RNNs are used in applications such as Text-to-speech conversion, stock price prediction, etc.

3.2.3.3 Deep Belief Networks

Deep Belief Networks (DBNs) are another class of deep learning algorithm that use probabilities and unsupervised learning to give predictions. Unlike other architectures, in DBNs, each layer gains knowledge from the input and works globally and regulates each layer. In this type of network, every node is connected to their prior and their subsequent layers. While their initial nodes are undirected towards the hidden layers, subsequent layers are having directed connections. In deep belief networks, a "greedy algorithm" is implemented to pre-train the model, thus allowing the model to find the global optimum in a seamless manner.

3.2.3.4 Autoencoders

Autoencoders belong to a unique class of deep learning architectures. They are unique because the output obtained in this network is the same as input, i.e., the dimension of output is equal to that of input. Although the dimensions are the same, in the hidden layers which have a reduced number of nodes compared to the input layer and output layer, the input is reduced to its smaller representation before reconstructing the same in the output layer. Autoencoders consist of three components: Encoder, Code, and Decoder. An encoder is a feed-forward network which reduces the dimension of the input. This reduced representation is stored in the layer called code. This reduced representation is fed to the decoder which has a similar structure to that of the encoder. In this section, the reduced representation is reconstructed back to its original format. Some of the types of autoencoders are Convolutional Encoder, Variational Autoencoder, Denoising Autoencoders, and Deep Autoencoders. The following section explores some of the recent works on COVID-19 using Deep Learning.

3.2.4 Evaluation Metrics in Deep Learning

The performance of the CNN model has to be measured and should be compared against the CNN models that solves the problems of similar kind, In order to measure the performance of the CNN models, the following are the parameters used.

1. *Accuracy*- The accuracy of the model states how good the model is able to learn from the dataset supplied to it. The accuracy term is always expressed in terms of percentage. There are two types of accuracy when it comes to CNN models, they are
   a. *Test accuracy* – which gives the accuracy of the model during training phase
   b. *Validation accuracy* - which gives the accuracy of the model when a new dataset is provided for the classification or detection
2. *Loss* – The loss metrics is the converse of the Accuracy parameter; it gives the difference between the actual prediction of the model with expected prediction. The loss can be defined on the same terms of accuracy, i.e., Validation loss and testing loss that is the loss incurred during testing and validation phases of model training
3. *Precision* – It measures how many observations predicted by the CNN model as positive are in fact positive, it is also called as Positive Predictive value (PPV)
4. *Sensitivity* – It measures true positive rate.
5. *F-Score* – It is a weighted value of precision and recall.

4 Recent Works on COVID-19 Using Deep Learning

The following section summarizes some of the recent literature done on COVID-19 classification using novel or pre-trained convolutional neural networks. The following section examines a plethora of CNN architectures which are relevant to understand the recent research work done by various researchers around the globe. Some of these architectures such as [8, 16–18] are the CNN models which are custom developed and having implementation for the COVID-19 detection and classification. The other architectures used Transfer learning paradigm wherein, a pretrained model used for the similar task is reused again with tweaks on the layers to suit it to the COVID-19 Identification and Classification, these works can be referred here [10, 13, 20]. There are other architectures which are discussed further in the section which sheds some insights on the hybrid models which use a different approach by cascading architectures and techniques such as in [21, 23]. The surveyed works are summarized based on the following factors/parameters for COVID-19 detection and classification.

1. Type of Dataset
2. Pre-processing methodology,
3. Novel CNN architectures,
4. Architectures using Transfer Learning,
5. Binary or Multiclass Classification,
6. Other CNN architecture

4.1 Type of Dataset

From the radiological perspective, most of the work that has been carried out by various researchers use radiology datasets such as chest X-ray, chest CT scans, etc. for their experiments using Deep learning. The work carried out by [8] used a chest-X-ray dataset to train the modified DarkCOVIDnet. The work presented by [10] proposed a novel CNN architecture called CovXNet which is trained on the Chest-X-ray images which were obtained from the Guangzhou Medical Center, China [11] which comprises of chest X-ray images of COVID, Normal, Pneumonia (viral), pneumonia (bacterial) patients and is signed by the radiologists. COVIDiagnosis-Net [15] is another novel work that uses Deep Bayes-SqueezeNet based technique to provide diagnostics about coronavirus using the COVIDX Dataset which comprises posteroanterior chest radiography images for training the model. Though there is a wide range of open-source datasets available for radiology datasets for chest CT scans and X-Ray, many of the images in the datasets are collected raw and need additional steps of preprocessing before usage. These datasets are constantly updated from various sources regularly and the updated datasets can be used for further tuning of various models. Out of all the datasets, the COVIDX dataset is the biggest one in the numbers of COVID-19 positive cases.

4.2 Pre-processing Methodology

In order to train a custom CNN or to use an existing pre-trained model either using it as a stock or for fine-tuning, the input data to the CNN's must to be preprocessed such that it complies to the existing pre-trained architecture guidelines. Some of the techniques involve the conversion of RGB images into Gray-Scale images which helps to speed up the processing of images in CNN adding to the advantages such as ease of visualization in grayscale, lesser space requirement, etc. The other form of preprocessing methods involves applying Fourier or Discrete Wavelet Transforms on the images and feeding the processed images for training in CNN [12]. Each of the CNN mandates the range of input image size to be used for their respective architecture. Theoretical understanding of the pre-trained architectures is needed to decide on the exact image size to which the input images need to be resized. For example, the publicly available dataset presented in the chest X-ray image dataset [9] consists of a wide range of images at different sizes and resolutions. The CNN requires the image to be resized to fit into CNN for training. The work presented in [8] uses resize operation on the dataset using python tools and API's such as OpenCV [9] and resizes the image to 256×256 size and used for training the model. Similarly, different models need the images to be fed at a specified size such as the work [10] preprocess the images in the dataset to 128×128. The other notable preprocessed image size is 224×224 which is used in [15]. The uniformity of the image size is quite pivotal in the CNN model performance. The method of generating additional images for training can be achieved with the concept of augmentation where the images are rescaled, sheared, and moved randomly in all directions and considered as the additional dataset for processing. These kinds of operations are vital in training CNN with limited datasets. The work presented in [15] used data- augmentation approach to generate additional data for the training and validation. The following Fig. 8 depicts some of the preprocessing operations employed in CNN.

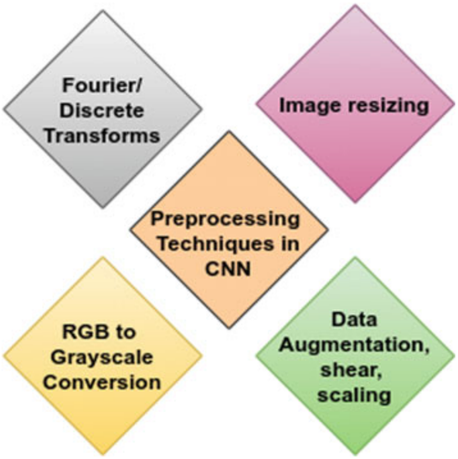


Fig. 8 Data preprocessing techniques in CNN

4.3 Novel CNN Architectures

The following subsection explores some of the novel architecture which is used for COVID-19 detection and classification.

4.3.1 DarkCOVIDNet

The architecture presented in [8] is called DarkCOVIDNet. The model designed yielded 98.08% accuracy for binary classes and 87.02% accuracy for multi-class classification. The dataset used for the model is taken from the combined dataset accumulated by Cohen JP [9] using various opens access sources. The work uses the DarkNet-19 pre-trained model which is used in a real-time object detection system named YOLO (you only look once), basically used for classification. The proposed architecture for basic classification is as shown in Fig. 10.

The model was initially trained for three classifications, which are COVID1, No-findings, Pneumonia and then later the trained model was tested against two cases COVID, No-findings. The model performed pretty good for the dataset and was able to attain an accuracy of 98.08%. The findings are plotted in the graph and can be seen in Fig. 9. Though the accuracy seems valid, the nature of the model learning is erratic, an apparent steep fall in the accuracy value at certain points in the plot is evident from Fig. 10. Further tune-up of the model is possible which might smoothen the learning of the model about COVID-19 features.

The heart of the architecture is the usage of object detection architecture such as DarkNet-19 as a base architecture. The key takeaway from the work is the fact that prebuilt object detection CNN's can also be used for Detection of COVID-19 from chest X-ray images assuming that the pattern of infection in the chest X-ray images can be treated as objects.

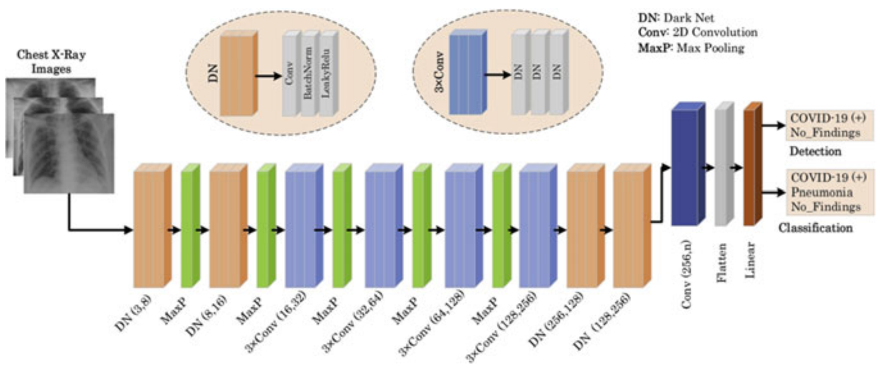


Fig. 9 The architecture of the DarkCOVIDNet [8]

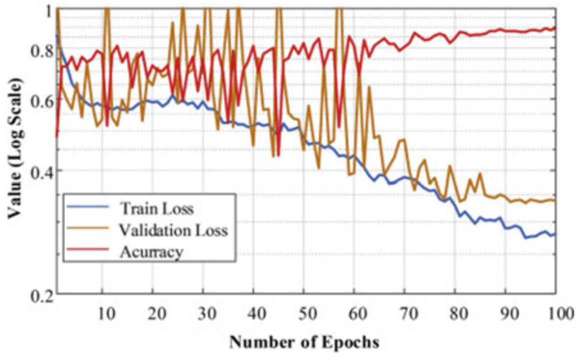


Fig. 10 Results obtained by the DarkCOVIDNet [8]

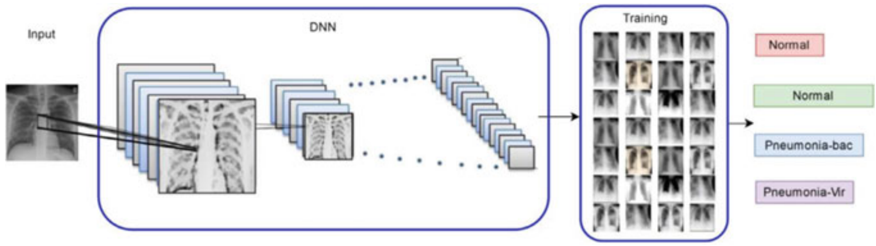


Fig. 11 The architecture of CoroNet [14]

4.3.2 CoroNet

The work presented in [14] introduces a new CNN architecture which is based on the Xception architecture, which is a classification model, pre-trained to process the general images from Imagenet. The model is applied with COVID-19, chest pneumonia X-ray images from publicly available databases [9].

It is a 4-class classification that is COVID/pneumonia bacteria/pneumonia viral/Normal) and they achieved an accuracy of 89.6%. For three-class classification (COVID/Pneumonia/normal) they attained the accuracy of up to 95%. The architecture of the model is as shown in Fig. 11.

CoroNet is an architecture that is custom tweaked for the identification and classification of COVID-19. The architecture uses Xception CNN architecture as its base model [20] and it comprises 71 layers deep CNN model which is trained using the ImageNet image dataset [10]. The Xception model uses depthwise separation of the convolutional layers and it uses residual connections in place of regular convolutional layers. This reduces the number of operations required for the classification since the depth wise separation converts $n \times n \times k$ convolutional operations into $1 \times 1 \times k$ operations which in turn reduces the convolutional operations by 1/k. The residual

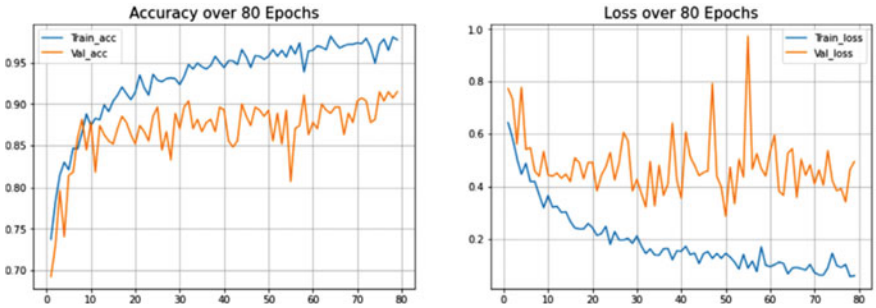


Fig. 12 CoroNet performance in terms of accuracy and loss for epoch = 80 [14]

connections are those connections that allow certain convolutional layer output to skip over a few layers. The networks which employ these are called residual networks.

The idea behind the usage of residual networks is to overcome the problem of vanishing gradient. The results obtained by the model are as shown in Fig. 12 and the dataset used for the training of this model is taken from [9].

With the residual networks in place, the CNN designed used very few layers for the initial learning of COVID-19 features from the dataset. This had speed up the overall learning time needed for the Model to learn the patterns of COVID-19 features indeed. The model also sheds some insights about the execution time which is comparatively less due to the use of Residual networks.

4.3.3 COVIDiagnosis-Net

The work in [15] introduces a COVID-19 detection model based on Artificial Intelligence. The model uses deep SqueezeNet with Bayes optimization techniques for attaining higher accuracy. The dataset used for training is from [16] which is a COVIDx dataset. The architecture addresses the irregularities or noises in the images in the dataset by performing offline data augmentation and then the processed images are fed into the training steps of CNN. The offline augmentation enables the model to be used to build apps on handheld devices or embedded systems as well. The Augmented data is then passed through the standard SqueezeNet pretrained Model. SqueezeNet architectures are designed to have very few layers with lesser datasets which helps to easily fit into computer memory that can be transmitted over the networks. The work also uses hyperparameter tuning as a feedback network and adjusts the weight accordingly with the help of the Bayesian optimization method. The detailed architecture of the COVIDiagnosis-Net is as shown in Fig. 13.

The model is trained with a dataset [16] and the model yielded an accuracy of 98.3% with the F1 score of 0.983. The model uses offline methods to do the preprocessing of the images which reduce the overall workload of the designed architecture. The core of this architecture is the offline augmentation method combined with

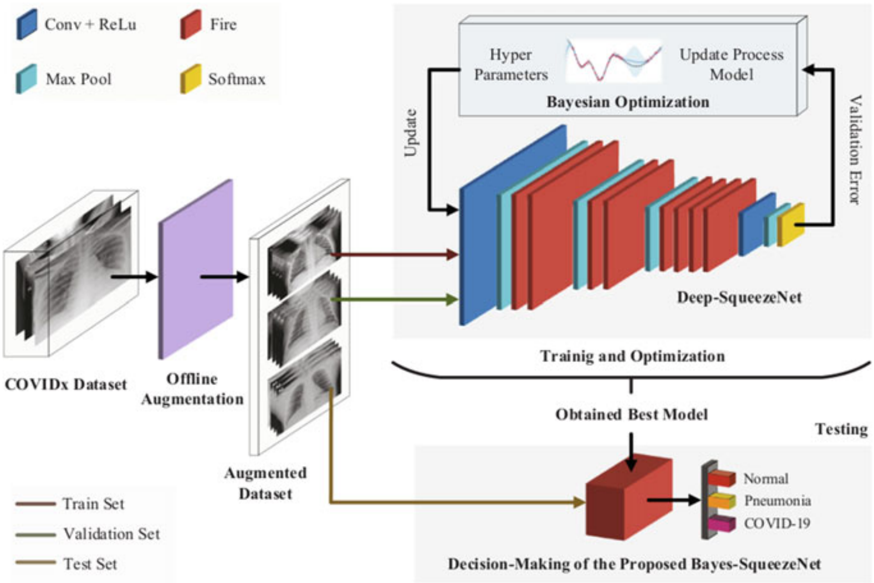


Fig. 13 COVIDiagnosis-Net architecture [15]

Bayesian optimization which is used to optimize expensive to evaluate functions which requires further exploration to be used for new architecture designs.

4.3.4 COVID-Net

Another important architecture that was trained on the COVIDx dataset as presented in [16]. The model is custom-tailored for COVID-19 Identification and Classification.

The model is designed for 3 Class Classifications, i.e. No infection (Normal), Non-COVID (viral/bacterial), and COVID infected. The detailed model of the COVID-Net architecture is as shown in Fig. 14. The model is compared against the VGG-19 and ResNet-50 architecture and it outperformed both in terms of accuracy. The model yielded 90.5,91.3 and 98.9 respectively for normal, non-COVID, and COVID infected.

The COVID-Net architecture is pre-trained with the ImageNet dataset and then was trained on COVIDx data. The architecture varies the learning rate over a period and uses Adam optimizer for learning. If the learning rate is stagnant for a certain period, the learning rate is decreased. The architecture also uses augmentation, residual networks, and hyperparameter tuning methods to attain the results. The key takeaway from exploring this architecture is the hyper-parameter tuning which can also be an effective method to improvise the accuracy and performance of the model. Some of the hyperparameters which can be tuned are, size of the network,

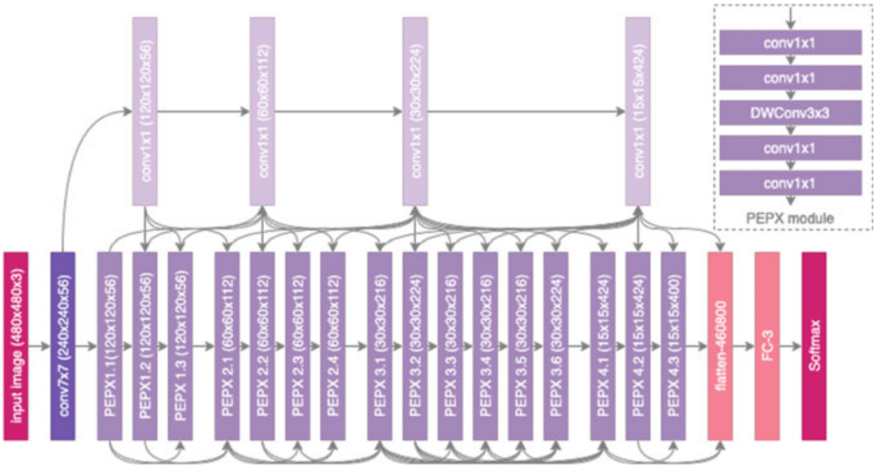


Fig. 14 COVID-Net architecture [16]

number of layers, dropout rate, learning rate, kernel size etc. By varying these parameters after the initial learning, the performance can be improvised over time which is quite evident from the COVID-Net.

4.4 CNN Architectures Based on Transfer Learning

Transfer learning is a paradigm using which the knowledge and learning gained from the one model is reused or transferred to another model for which the features and dataset being used for learning are similar. The approach is widely used because it is tedious to develop a model well suited for the application cause from scratch and it makes sense that if there is a pre-trained model which has been designed to solve a similar kind of problem earlier to be reused. This effectively reduces the time and cost involved in designing the model to solve the problem using the Deep learning approach.

When it comes to the reusability of the learning, there are many approaches, it depends on the type of the problem, dataset and its features, and the approach itself. Some of the approaches are, ***Classifier, Standalone Feature Extractor, Integrated Feature Extractor, Weight Initialization and update***. The following subsections explore some of the architectures which are used for COVID-19 detection and classification using the transfer learning approach.

4.4.1 CovXNet

The work proposes a novel CNN architecture termed as CovXNet which uses dilation rate and depth wise convolution for extracting the rich features out of the chest X-ray [10] as shown in Fig. 15. COVID-19 does cause pneumonia and the symptoms of it are kind of similar to normal pneumonia, the model is trained first with X-ray images of the (viral/bacterial) pneumonia. The trained model weights are then transferred using the concept of Transfer learning and several layers are fine-tuned to process the X-ray images. The model is trained with X-ray images of infected/non-infected patients of pneumonia for various resolutions and each of the learning was accumulated in the end for the meta-learning process with the help of the Stacking Algorithm. The results obtained were having an accuracy of 97.4% for COVID.

The detailed architecture is as shown in Fig. 16. The study also experimented with the models which are trained with resized images at various levels and used the learning from each model to perform output prediction. The process is termed as a Meta-learning phase where the inferences from the models are accumulated to

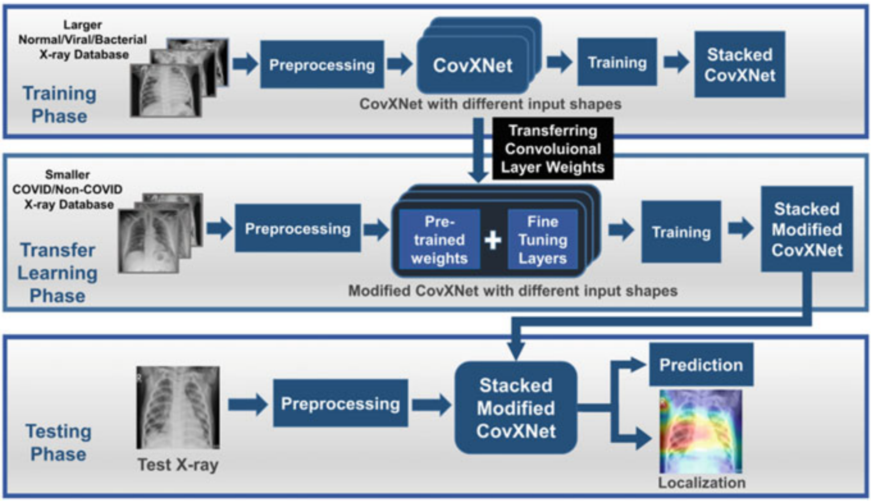


Fig. 15 The workflow of CovXNet [10]

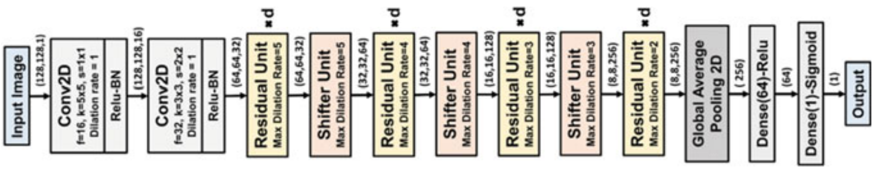


Fig. 16 The architecture of CovXNet [10]

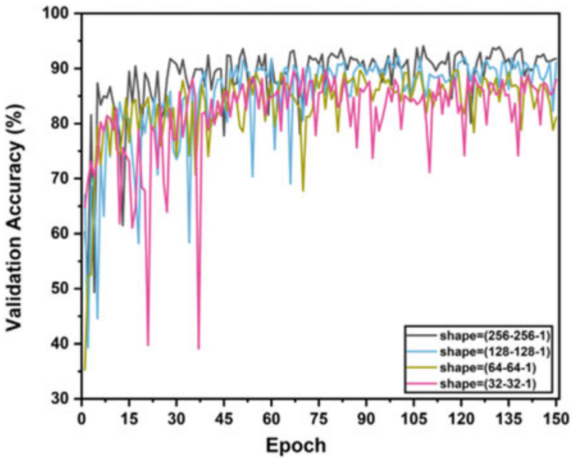


Fig. 17 Epoch vs accuracy for stacked meta learner architecture [10]

a final summative learning model called Meta Learner which does the predictions. The results obtained are summarized below in Fig. 17.

4.4.2 nCOVnet

The work proposed in [13] introduces a new CNN architecture that uses the Transfer learning approach. The Model is called nCOVnet which is used to detect the COVID-19 infection based on the chest X-ray images. The model uses a dataset compiled by Cohen et al. [9] for training and validation. The model uses VGG-16 as base model to which custom 5 layers are added and the modified VGG16 model was trained on the dataset [9]. The input images of the model are of size 224 × 224 × 3 which is of depth 3 and an overall image dataset of 337 images is used to train the model. The model provided an accuracy of 97% for COVID-19 positive patients and an overall accuracy of 88%. The model was trained for 80 epochs with the learning rate to be 0.0001. The training accuracy yielded was around 93–97%. Once the model was trained and weights were computed, the model boasts to take a mere 5 sec for COVID-19 classification on the supplied new images. The results obtained are presented in Fig. 18.

4.4.3 Automated Deep Transfer Learning Approach for COVID-19 Detection

The work presented in [17] used the Xception model as a base model for feature extraction and the learning was transferred to the custom model for classification of COVID-19. The overall model is depicted in the following Fig. 19.

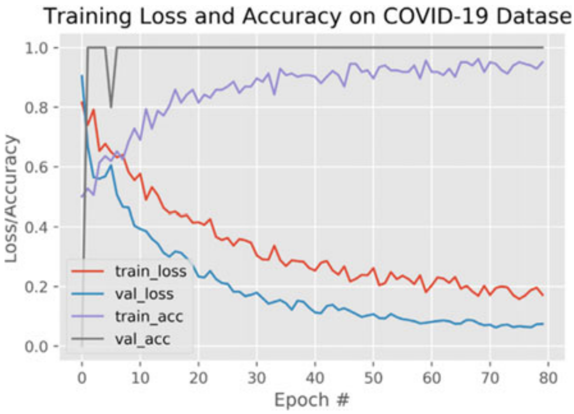


Fig. 18 Loss/accuracy vs epoch stats of nCOVnet [13]

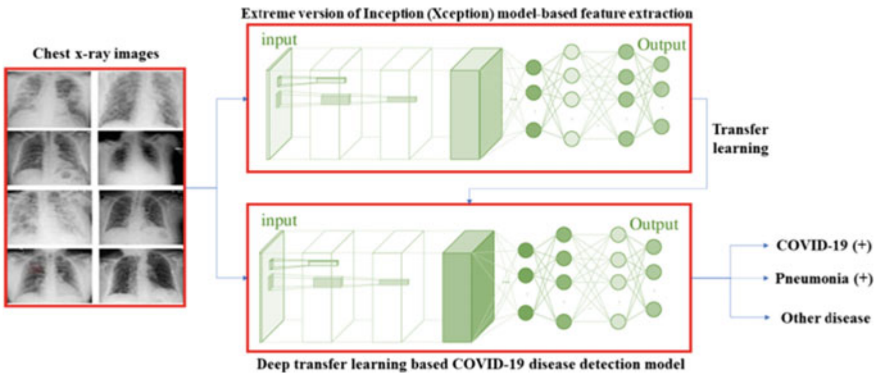


Fig. 19 The architecture used for the classification of COVID-19 [17]

The model was trained on the chest image X-ray dataset [8, 9]. The model uses an additional logistic regression technique to perform the binary classification i.e., COVID-19, and non-COVID. The model obtained an overall accuracy of 99.2% in training and 97% accuracy during validation.

The concept of Transfer Learning is a very effective approach because it saves time of building the model from the scratch for COVID-19 detection and classification. There is plethora of algorithms which have worked effectively for the general image classification whose learning can be reused to solve the problems related to COVID-19 classification. The transfer learning is actually used as either weight initialization approach or as a feature extractor. The architecture of Modified CovXnet embodies the first approach of weight transfer which is an executive summary of the learning from the pre-trained model called as CovXnet. The architecture called nCovNet [13] also uses the weight transfer paradigm for transferring the learning of the model for COVID-19 detection and Classification. The work proposed in [18] demonstrates the feature transfer from the final Neural Network output and input it as another input to the layers of the Second Model which is to be trained on COVID-19 dataset. The approach seems to be working as it was evident from the performance parameters of [18]. The approach has to be examined carefully, both the approaches might work or might not work for various architecture which is another research horizon to explore.

4.5 Binary v/s Multiclass Classification

Another horizon using which the literature survey was conducted is based on whether the model is doing Binary classification or Multiclass classification. While examining the Radiology data such as CT images, X-ray images of COVID-19 datasets from the various public sources for COVID-19 identification and classification, the features that were under focus for being examined by the various CNN models were similar to that of pneumonia disease which was induced by the other viruses or bacteria. Moreover, COVID-19 infection does also exhibit pneumonia-like features when examined using X-ray and CT images of COVID-19 infected patients. There were CNN models that were developed to address the false positive cases as well and it is summarized in Table 1 below.

Table 1 Classification of Deep learning models based on the mode of classification

Model name	Binary/Multiclass	Classes	Literature reference
DarkCOVIDNet	Binary & Multiclass	COVID/non-COVID (Multiclass) COVID/non-COVID/Pneumonia (Multiclass)	[8]
CoroNet	Multiclass	COVID/pneumonia(bacteria)/pneumonia (viral)/Normal (4-class) COVID/Pneumonia/normal (3-class)	[14]
COVIDiagnosis-Net	Multiclass	COVID/Pneumonia/normal (3-class)	[15]
COVID-Net	Multiclass	COVID/Pneumonia/normal (3-class)	[16]
CovXNet	Binary/Multiclass	COVID/non-COVID (Multiclass) COVID/non-COVID/Pneumonia (Multiclass)	[10]
nCOVnet	Binary	COVID/non-COVID (Multiclass)	[13]
Automated Deep Transfer Learning Approach for COVID-19 detection	Multiclass	COVID/Pneumonia/normal (3-class)	[17]

4.6 Other Notable Architectures Used for COVID-19 Classification

This chapter has presented a range of Deep learning methods for Covid 19 related classification problems. In this section we present some of the noteworthy Deep learning model which also concentrate on learning for Covid 19 related tasks.

4.6.1 Convolutional CapsNet

The work [18] proposes a novel approach called "capsnet" for the detection and classification of the COVID-19 pandemic. The core idea behind the work is the use of the capsule networks which provide fast and accurate diagnostics of the COVID-19. The model is tested for the binary classification as well as multiclass classification and achieved an accuracy of 97.24 and 84.22% respectively. The method can be used along with the regular diagnosis of the medical practitioners for fast screening. The architecture of the CapsNet is shown in Fig. 20.

Capsule networks are a type of Artificial Neural networks which are used to model the hierarchical relationship that exists in the training images. The idea behind the use of this network is to add structures called "capsules" with the regular layers of CNN and reuse the learning of these capsules to obtain the final learning or higher learning. The output of these capsules is a vector comprising the probability of observation and a pose for that observation [19]. In the convolutional neural network, the learned features are pooled in the pooling layers before transmitting to the next layer. The

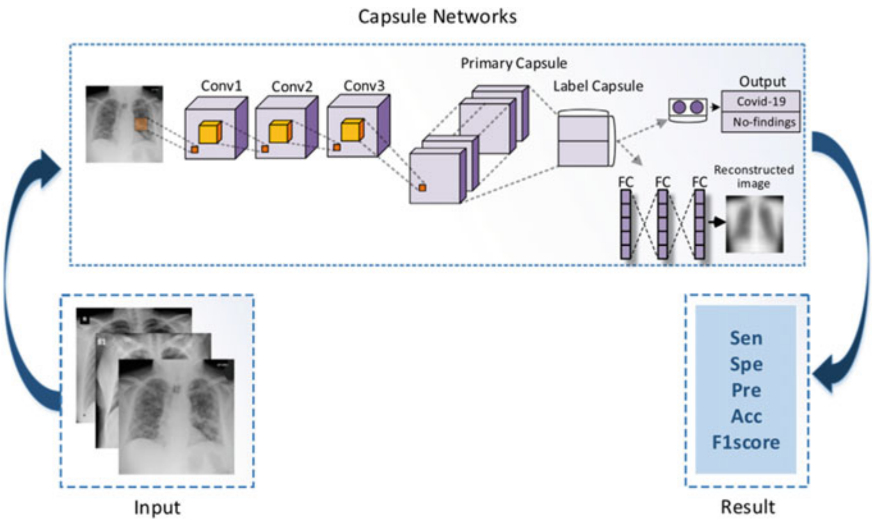


Fig. 20 Flowchart of the CapsNet [18]

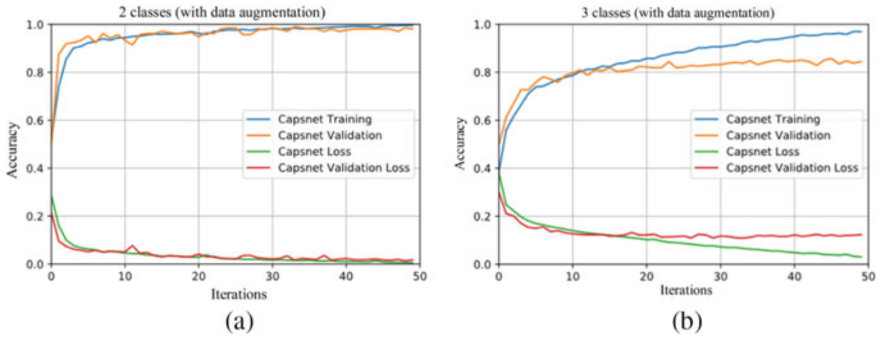


Fig. 21 **a** Accuracy vs epoch for 2 Class classification using Capsnet (with augmentation) (left), **b** Accuracy vs epoch for 3 Class classification using Capsnet (with augmentation) (right)

capsules can be used to trap this information before going to the fully connected layers and can be combined in the later stages to yield better accuracy. The model was trained on the X-ray dataset which was accumulated using [9] and other open source databases and it consists of 1050-COVID-19 images with 1050 non-COVID chest X-rays images. The approach uses a data augmentation approach for the data preprocessing to improvise the accuracy. The results are summarized in Fig. 21(a) and (b).

4.6.2 CNN Architecture by Combining Standard Models

The Table 2 summarizes the performance of the various CNN architectures used for COVID-19 Detection and classification.

The work presented in [20] proposes a different approach wherein the output of two CNNs are combined and passed on to the custom CNN model for COVID-19 detection and classification as shown in Fig. 22.

The concept itself is interesting to be further explored since the learning from two models might increase the performance of the output, but still careful considerations needs to be done before deciding on the architectures to be used. The hinderance of this approach is the time taken for the model design and parameters since the model requires the two models of choice to complete the learning before passing the features to the custom CNN architecture for the output. The various architecture performances with COVID-19 classification (Binary/Multiclass) is summarized in the following Fig. 23 , and Fig. 24.

Table 2 Architectures for COVID-19 detection and classification

Work	Dataset used	Transfer learning	Architecture name	Binary/Multiclass	Accuracy
[8]	[9]	NO	DarkCOVIDNet	Binary/Multiclass	Binary: 98.08%, Multiclass: 87.02%
[14]	[9]	NO	CoroNet	Multiclass	Multiclass: 89.6 (4 class)–95% (3 class)
[15]	[16]	NO	COVIDiagnosis-Net	Multiclass	Multiclass: 98.3%
[16]	[16]	YES	COVID-Net	Multiclass	Multiclass: 91.3%
[10]	[10]	YES	CovXNet	Binary/Multiclass	Multiclass: 90.2%
[13]	[9]	YES	nCOVnet	Binary	Binary: 93%
[17]	[8, 9]	YES	Automated Deep Transfer Learning	Multiclass	Multiclass: 97%
[18]	[9, 16]	NO	CapsNet	Binary/Multiclass	Binary: 97.24%, Multiclass: 84.22%
[20]	[9]	NO	Hybrid Architecture	Multiclass	Multiclass: 99.5%

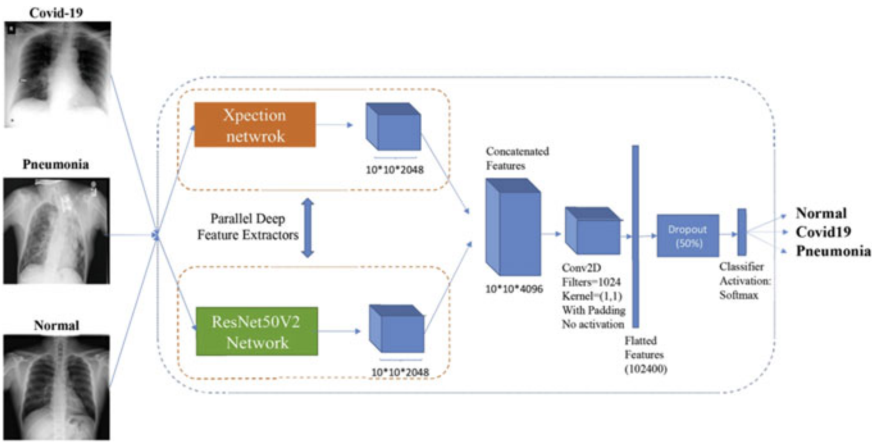


Fig. 22 The architecture of the Hybrid Approach using Xception and ResNet50V2 [20]

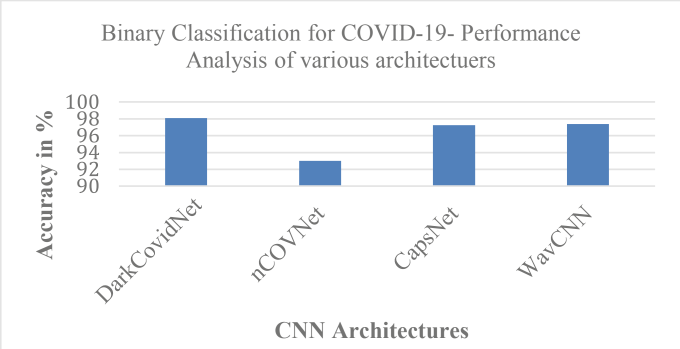


Fig. 23 Binary classification performance analysis of various architectures for COVID-19 classification

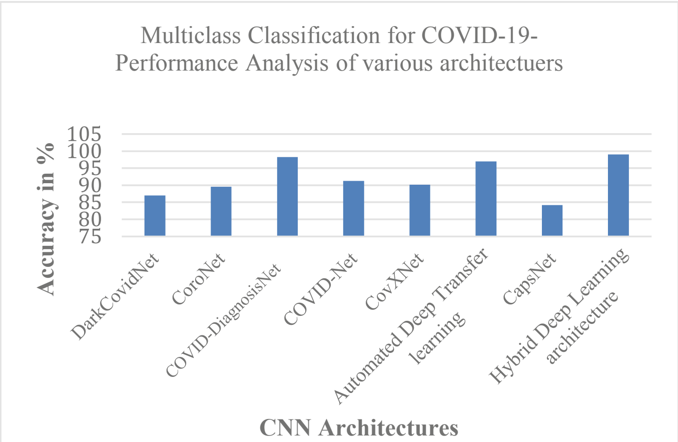


Fig. 24 Multiclass classification performance analysis of various CNN architectures for COVID-19 classification

5 Sources of COVID-19 Datasets

The following section examines the various datasets that are used by various researchers all over the world for the COVID-19 detection and classification. An extensive detailed survey of various COVID-19 datasets was conducted by the researchers covering the various data sources and the following table summarizes the various datasets, its sources, and applications [21] as shown in Table 3.

The datasets quoted above are all available as public for the usages of the researchers, some of these datasets are compiled using the existing datasets. In terms of the number of samples for chest-X-ray COVIDNet stands out with 13,800 chest

Table 3. COVID-19 X-ray and CT image dataset (chest/lungs)

Dataset name	Type	Description	Learning architectures	Learning goals	Source
COVID-chest X-ray-dataset	Chest X-ray	(Total Samples-654) COVID-19 Chest X-Ray (203)	Miscellaneous	Data collection	[9]
COVIDx	Chest X-ray	Normal (8066), pneumonia (5575), COVID-19 (617), Total - 14258	COVIDNet,	COVID-19 identification and classification	[27]
Extensive COVID-19 X-Ray and CT chest images dataset	Chest X-ray/CT images	X-ray images (5500-Normal, 4044-COVID-19), CT images (2628-Normal, 5427-COVID-19 infected)	Miscellaneous	Data collection	[26]
COVID-19 Radiography database	Chest X-ray	COVID-19 (1143), Normal (1341), Viral Pnemonia (1345)	Miscellaneous	Data collection	[28]
Miscellaneous	Chest CT images	349 CT images COVID	NA	COVID-19 identification and classification	[2]
COVID-19 SIRM database	Chest X-ray	COVID-19 (68 images)	Miscellaneous	Data collection	[29]

X-ray images which is an amalgamation of many image dataset from publicly available datasets. As far as the Chest CT image set is concerned, the CT image dataset that is used in [2] stands out with a greater number of samples. The image quality of the datasets specified in the table are substantial with standard quality and can be used by the researchers to carry out their independent research work.

6 Discrete Wavelet Transforms and CNN

The following section examines the impact of combining traditional image processing methods such as Discrete Wavelet Transforms (DWT) with CNN's using pretrained models as a base. The section also examines the concept of DWT in detail with relevant examples. The outcome of this study is to examine the impact of DWT when combined with CNN's for solving classification problem. Though efforts have been done to use DWT along with CNN's for modelling problem such as image reconstruction, texture mapping, frequency observation, etc., not much substantial work is done on the usage of DWT in addressing classification problem.

The two approaches of transformations that can be applied on the images for image processing paradigm such as classification, object detection, tracking, compression, restoration, segmentation, etc. are, Spatial Approach, Spectral Approach.

6.1 Frequency Domain Processing

The term “spectral” means frequency, the spectral Approach relies on the modified spatial images which are obtained upon applying Fourier transforms on it. Spectral Image processing is very vital in technology such as Image reconstructions, object tracking, Automated target detection [24], Filtering and Compression etc. Some of the spectral approaches use either Fourier or wavelet transforms over spatial images. The following subsection will explore each of them in detail.

6.1.1 Discrete Wavelet Transform on Images

Wavelets (small waves) are the functions that are concentrated in time as well as in frequency around a certain point in the image. Wavelet transforms rely on the low and high pass filter extensively to gain insights of the various frequency levels along with their locality. The method is called Discrete Wavelet Transform because the method is focused on a range of frequencies in the image rather than the entire set of frequencies.

In [25], Parida, P., and Bhoi, N. discuss the discrete wavelet transformation method as shown in Fig. 25. Which gives a view of how DWT extracts the features by applying low and high pass filters at the cost of resolution due to subsampling.

The filters are applied column-wise along the image, upon the application of the filter, there will be redundant samples coming out of the low pass and high pass filter.

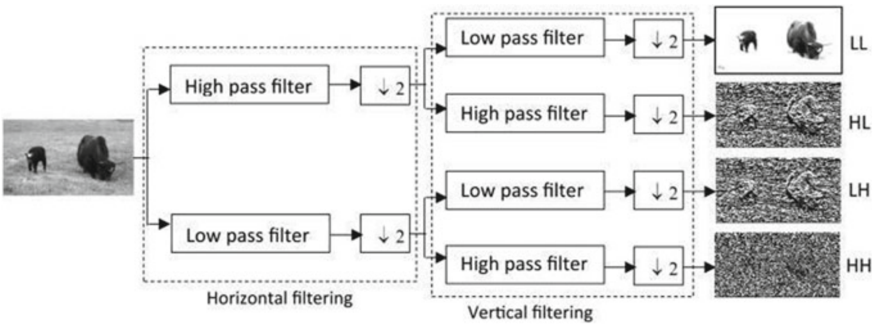


Fig. 25 Discrete wavelet transform applied to a sample image [25]

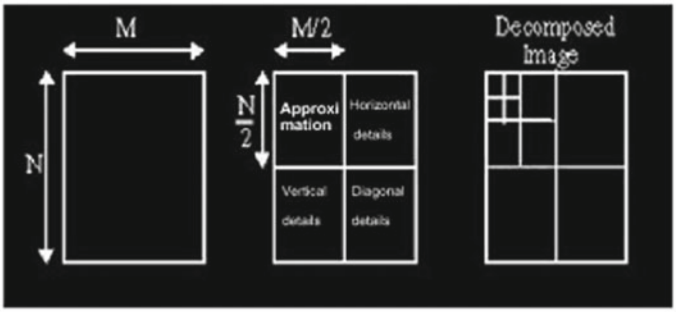


Fig. 26 Level 1 decomposition [25]

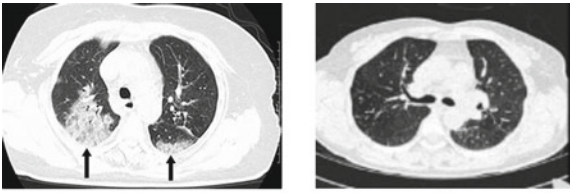
Since losing half of the samples would not let one lose information, Down sampling operation is done on the image by 2 every iteration (Nyquist Algorithm). The output of the low pass filter and high pass filter after sub-sampling is fed to the low pass and high pass filter again followed by a subsampling by 2, the difference lies in the fact that the operation is applied row-wise on the image. The output obtained at the end of this iteration will be an image that passed through LL (2 low pass), LH (Low pass followed with high pass), HL (High pass followed with Low pass) and HH (2 High pass filter). One important concept to be aware of is that High pass filter extract the edges and low pass filter does the approximation of the image. The output after this is represented as Fig. 26 shown below.

The LL image gives the Approximation, HH gives diagonal edges (since it passes through two High pass filters, first with column and with the row), LH gives the horizontal edges of the image and HL gives the vertical edges of the image. Some of the important observations from the study are, in this process, the images are sampled twice, once for row and once for column incurring sampling rate of 1/4th the image. The resolution of the image also decreases at the same rate. The overall purpose of this DWT is finding the localization of frequency as well as locality, the study helps us to understand that in a particular location whether we have higher or lower frequency which will help us to understand the distribution of frequency over a specified portion of the image.

a. Experimental Testbed

In order to study the behavior of the CNN models with DWT in place, a pre-trained model named 'AlexNet' which is pretrained for general image classification using a subset of dataset from ImageNet dataset (~1.2 Million Images-subset) and it has classified objects among 10,000 classes approximately. The dataset comprises of 349 CT images found from clinical findings of COVID-19 from 216 patients and 397 Non-COVIID CT images.

The overall objective of the study to train the AlexNet to learn the frequency distribution of the infection from the Chest CT images of COVID/Non-COVID image set. The following diagram shows the progression of the COVID-19 infection 12 days after the contact of the virus as shown in the Fig. 27a, the Fig. 27b shows the CT



(a) (b)

Fig. 27 **a** - chest CT view of the COVID-infected patient. **b** - normal chest CT view of Non-COVID patient

Parameters	AlexNet	
Layer 1 (CL)	Fliters	96
	Kernel Siz	11*11
	Padding	valid
	Stride	4
	Activatio	Relu
Max Pooling	Poolsize	2*2
	Stride	2
	Padding	valid
Batch normalization		
Layer 2(CL)	Fliters	256
	Kernel Siz	11*11
	Padding	valid
	Stride	4
	Activatio	Relu
Max Pooling	Poolsize	2*2
	Stride	1
	Padding	valid
Batch normalization		

Layer 3(CL)	Fliters	384
	Kernel Siz	3*3
	Padding	valid/1
	Stride	1
	Activatio	Relu
Batch normalization		
Layer 4(CL)	Fliters	384
	Kernel Siz	3*3
	Padding	valid/1
	Stride	1
	Activatio	Relu
Batch normalization		
Layer 5(CL)	Fliters	256
	Kernel Siz	3*3
	Padding	valid/1
	Stride	1
	Activatio	Relu
Max Pooling	Poolsize	3*3
	Stride	2
	Padding	valid
Batch normalization		

Basic Parameters		
Input shape	224X224X3	
Output activation functio	Softmax	Sigmoid

Other parameters		
loss model(crossentropy)	categorical	Binary
optimizer	Adam	
Metrics	Accuracy	
Batch Size	10	30

Fig. 28 Model parameters of the AlexNet architecture (implemented)

image of the non-infected patient, and the distribution of the infection is pointed out in the Fig. 26a.

The experiments were carried out in two phases, Phase-I—Use the stock AlexNet Architecture without any preprocessing step apart from the resize operation and record the behavior, Phase-II Use the DWT preprocessing and feed the AlexNet with the preprocessed images on various layers with various inputs passing through the filters such as LL/LH/HL/HH.

6.1.2 Phase-I Implementation and Result

The Stock architecture of AlexNet [3] is fed with the input dataset comprising the 349/397 COVID/Non-COVID images from [2].

The model parameters used for the experimentation is as shown in the below Fig. 28. The architecture uses the stock parameters for the model design, and it

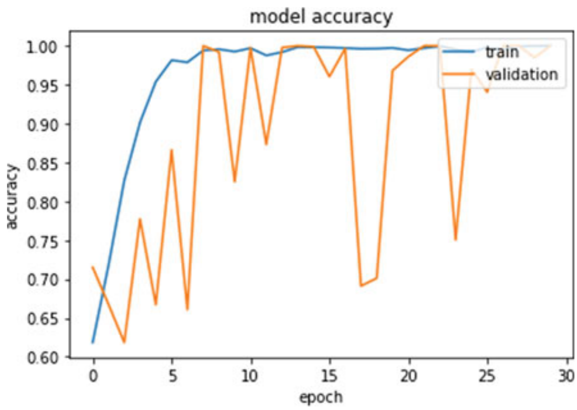


Fig. 29 Accuracy v/s Epoch Stock AlexNet architecture

is implemented using Keras Library. The result of the Experiment shows the erratic nature of the model during validation. The graph of Accuracy v/s epoch for the Stock AlexNet architecture shows the same on run. The Fig. 29 shows the graph. From the experiment, the AlexNet appears to have an erratic learning when it comes to the medical images especially the COVID-19 images and its quite evident in the accuracy value during validation. This is due to the nature of the input; the medical images are rich in information and are taken under the restricted environment. More ever additional model tweaking of the parameters such as Optimizers, Pooling Layers, loss models, or increasing dense layers did not induce the smoothness in the validation curve in the Graph.

6.1.3 Phase-II Implementation and Result

The following Fig. 30 shows the model design for the AlexNet with Discrete Wavelet Transforms as preprocessing method, the wavelet used for the transforms is biorthogonal wavelet.

The idea here is to make the AlexNet remember these distributions of the frequencies from the COVID-19 dataset and use the learning for the Classification of the new input images. Though the preprocessing steps involve the four-layer output, the HL and HH layer output is not of significant importance, so the Model is focused on the approximation of the original image (LL) and Horizontal edges (LH) processed images. The output of the DWT is as shown below Table 4. The AlexNet is first trained with the LL images (Preprocessed) with the stock architecture under standard settings, the results obtained is as shown in the Fig. 31. The state-of-the-art performance of 98% is achieved within first 15 epoch and is evident from the diagram. The inference drawn out of this is the fact that the AlexNet was able to learn the features using the approximation of the images of Chest CT images (for COVID-19 infected) obtained by the LL preprocessing method.

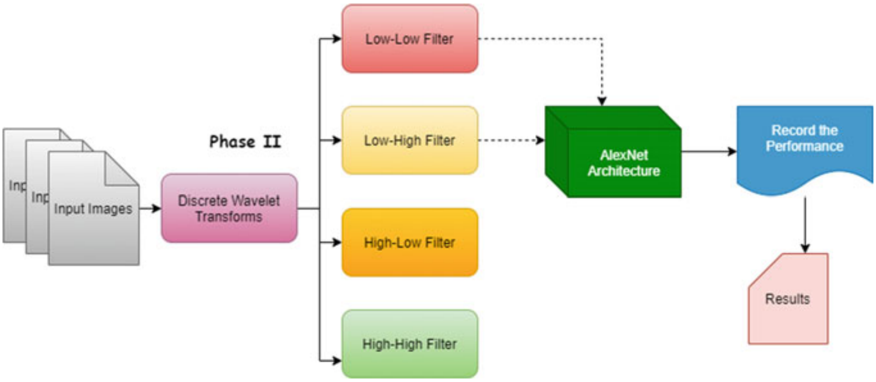
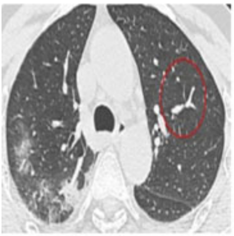
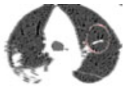


Fig. 30 Proposed architecture for experimentation

Table 4 DWT in action (chest CT image of COVID-19 infected patient)

	Low pass – Low Pass filter (LL)	
	Low Pass – High Pass Filter (LH)	
	High Pass – Low Pass filter (HL)	
	High Pass – High Pass filter (HH)	

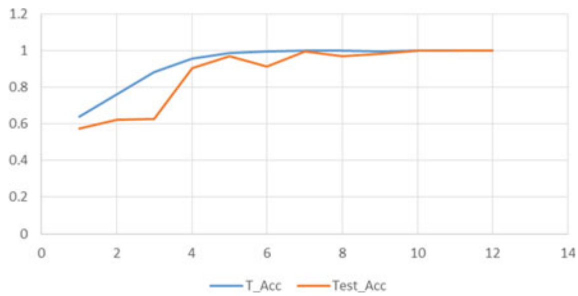


Fig. 31 AlexNet with DWT performance (accuracy vs epoch)

Though the features and image size were less, the validation curve appears to be smooth compared the erratic behavior exhibited by the AlexNet without DWT preprocessing. Upon applying the preprocessing of the images using LL1 the image size is reduced by 1/4th of the volume and feature loss is also evident. But with less features to learn, the AlexNet had performed well for the training and testing image sets.

The output of the LH1 though on the other hand did not provide significant learning for the model and it failed as the horizontal features were too minimal for the model to learn. The output was furthered smoothened with the additional resize operation on the LL images as shown in the Fig. 31, The state-of-art results are obtained with minimum epochs which is evident in the performance graphs shown in Fig. 30 and 32.

Though the DWT preprocessing step has given the ability to the AlexNet to perform better in terms of accuracy and loss, the method involves subsampling of the images on every level of decomposition which might incur feature loss. The loss of features might also hinder the learning of models when more than two level of decomposition using DWT is done. Another important point to note is the fact that more than two level decomposition will not yield any results for any pretrained models. Though there are architectures in place for the COVID-19 detection which has yielded better results and some of which have gone through the radiological trials, the proposed architecture opens up a new horizon of building a hybrid model which combines the powerful techniques of the traditional image processing methods with the Convolutional Neural Networks and evidently it had made a significant impact on the performance of the model in terms of accuracy and loss metrics.

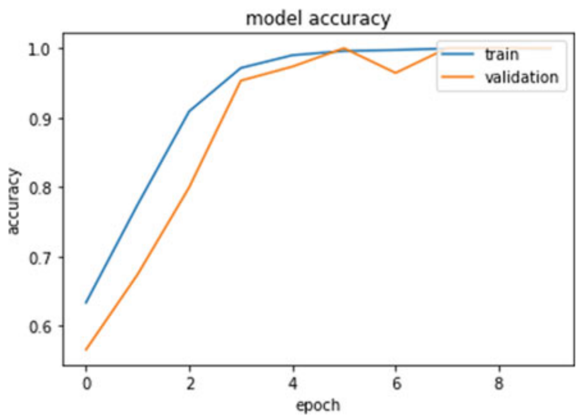


Fig. 32 AlexNet with DWT performance (accuracy vs epoch) with resize operation after the DWT step

Fig. 33 Research prospects

7 COVID-19 Medical Data Visualization

Medical information systems and medical devices are recording tons and tons of data related to treatment, disease progression, patient information, pace of disease spread, primary and secondary contracts etc. It becomes at most important to comprehend this data in an understandable way. The large quantity of data can be summarized with the help of data summarization and visualization tools. The summary and the visualization graphs generated will give the clear insight of what is happening on the ground, in turn helps in taking the informed decisions. It immensely helps the administration to act quickly, and reason the decision. There are several tools which could help in the process of summary generation and visualization. Following are some of the software tools that could benefit the healthcare community for visualizing and summarizing the humongous quantity of data currently being recorded in healthcare information systems.

Few of the noteworthy visualization of COVID 19 data include John Hopkins University Dashboard, WHO COVID-19 Dashboard, The University of Washington Dashboard, South China Morning Post Dashboard, CDC Dashboard, The COVID tracking project (COVID Tracking), ECDC Dashboard University of Virginia COVID-19 interpret and Few other attempts are COVID19-SEIR, COVID Trends, 91-DIVOC Rt Covid-19 Dashboard, IHME COVID-19 Dashboard, WORLDOMETER, COVID ACT Now Dashboard COVID-19 Measures Oxford, Carnegie Mellon Dashboard, Stanford COVID-19 Dashboard, and 91-DIVOC.

For in order to develop novel data visualization method for COVID 19, it is at most important to understand the pandemic demographic information, the disease trait and dynamics of the parameters. The new visualization method should not be generalized. From our observations at the outputs generated by the various visualization tools referred in the above paragraph, it is relatable to fact that the tools cannot be generalized to summarize/visualize from one population to other population, or in other words from one ethnic group to another ethnic group. A variable approach for the visualization of data needs to be implemented based on disease dynamic at any given place. We would surely recommend exploring this problem of visualization on the basis of demographics and ethnicity, our future work will definitely focus on this problem.

One other aspect visualization can explore is changing behavior of virus, depending on the weather pattern in different patterns across the world, one typical example could be - the European countries have gone to second lockdown due to high mortality rates, whilst in Asian countries (for example India), even though the infection rate is not under control, the mortality rate is significantly coming down. This contextual information about the disease dynamics needs to be considered rather mare generalizing the visualization approach. The visualization method can embody 360-degree attributes of the pandemic.

There are various aspects through which data can be visualized, but it is at most important to make sure that the visualization tool is not biased due to irrelevant assumption and generalization.

8 Factors that Indicate the COVID-19 Infection Progression (Radiology Perspective)

Among the radiology data such X-ray and chest CT images considered by the various researchers to develop CNN classification models, X-ray images tend to provide rich information related to the COVID-19 infection. The presence of patches or bands like ground glass opacity in mid-to-lower or peripheral lung zone in chest radiography is highly suggestive of COVID-19 infection. This observation along with the clinical observation can be collaborated to make final confirmation of COVID-19 infection [22]. Initial CT findings in COVID-19 show a similar kind of presence of patches in the lung zone of CT. The appearance of these patches is more in the lower lobes and is less frequent in the middle lobe. Some of the uncommon but observed findings are pericardial effusion, lymphadenopathy, cavitation & CT halo sign with the disease progression which is evident from the Chest CT scan images [23].

9 Framework and Research Methodology

The following Table 5 summarizes the survey framework used for the study.

Table 5 Survey framework

Objectives	1. Literature collection on the recent works done for COVID-19 detection 2. Literature survey on the dataset discovery and aggregation 3. Explore the CNN architectures, machine learning algorithms for the techniques for COVID-19 detection and classification 4. Explore the architectural designs and patterns for basing the proposed work
Survey methods	Systematic Exploration of articles, white papers, journals and medical databases/journals
Databases explored	IEEE-Xplore, Elsevier Journals, Radiopedia.org, Springer databases, acs.org, GitHub databases
Datasets accumulated	[2, 9, 16]
Keywords	Deep learning for COVID-19, COVID-19 classification and detection, Convolutional Neural Networks, Transfer Learning, Ensemble Learning
Inclusion criteria	COVID-19 Classification using CNN's
Exclusion criteria	Machine Learning for COVID-19 and its relevant works
#papers collected	127
#papers shortlisted	22
Outcomes achieved	1. Architectures summary presented in Table 2 2. Datasets examined under survey is presented in Table 3 3. Analysis of each architecture is presented under the architecture name in Sect. 4

10 Future Research Prospects

COVID 19 has posed several challenges and opportunities at a short span. Solving these challenges will yield greater benefit for the research fraternity at large. This section briefs some of the pressing challenges surfaced due to the pandemic situation, any researchers willing to take up COVID 19 related research topic can further explore the pointers given in this section. Each one of the pointers given in the rest of the sections can be explored in depth and suitable solutions can be proposed.

Data Science related areas could actively contribute to find out the solutions for threats developed due to infection, severity, and outcome risk. Infection risk is related to groups of people and the most susceptible set of individuals in that group. Severity risk is related to a specific or group of individuals developing the severe disease symptoms and complications. Outcome risk is related to the treatments and its outcome.

There are various realms from which the challenges related to COVID 19 looks challenging, if effectively solved could serve medical eternity in a larger extent.

The aspects to be considered for enhancement range from time complexity, space complexity, data quality, recording, record keeping, transfer learning, and in the aspect of transition networks. All these aspects need to be critically brought under the scrutiny of future research exploration.

There are some other aspects too from which the problems need to be solved, there needs to be methods and techniques which facilitate the way for faster development and discovery of drugs. It is more important to find out if the already available drug can cure the new disease. For this to happen, one should know the ontology of disease and the characteristics of the new disease. There are other dimensions which involve understanding the path of the virus, in other words, or to trace the virus entry point. Some other important aspects are Screening patients using face scans. Building biomedical knowledge graphs, predicting drug-target interactions, Predicting the spread of infectious disease using social networks, understanding viruses through proteins, Predicting viral-host protein–protein interactions.

11 Conclusion

It is apparent from the pandemic situation that, no country on this planet was prepared enough to handle the disaster. Unimaginable number of casualties have happened in such a short span of time, the effect of the disease spread has affected every individual in one or the other way. The economy is struggling, many have lost their livelihood, borders are closed, economic activities have come to halt, healthcare facilities and infrastructures are in a loan that have never been experienced in any time before.

It is a very challenging time that data science can come into action and solve some of the pressing issues. This paper presented an effort to consolidate the novel Deep Learning architectures for solving COVID 19 related problems. Several Deep Learning approaches have been proposed to address the problem of disease identification, progression, and diagnosis. This work also focused on comparing the several Deep Learning architectures based on several key performance factors. The attempt also was made to compare the different datasets used for evaluating Deep Learning architectures. The chapter also presented the need of Deep Learning instead of conventional neural network architectures. The concluding sections of the paper focused on the tools and techniques for visualizing and summarizing COVID 19 related data. Visualization and Summarization greatly helps in understanding the trend which facilitates informed decision making. There are several pressing issues in effectively handling the disease spread, diagnosis, and management - noteworthy issues related to pandemic which requires further investigation and research are also presented in the final sections of this chapter. The systematic attempt has been made to consolidate, compare, and evaluate the existing noteworthy Deep Learning architectures for handling various scenarios of COVID 19.

References

1. What Is COVID-19? | coronavirus. (n.d.), https://coronavirus.dc.gov/page/what-COVID-19. Accessed 28 Aug 2020

2. X. Yang, J. Zhao, Y. Zhang, S. Zhang, P. Xie, COVID-CT-Dataset: A CT Image Dataset about COVID-19 (n.d.), https://www.medrxiv.org/. Accessed 10 Nov 2020
3. A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks. Commun. ACM **60**(6), 84–90 (2017)
4. WHO Coronavirus Disease (COVID-19) Dashboard | WHO Coronavirus Disease (COVID-19) Dashboard (n.d.), https://COVID19.who.int/. Accessed 28 Aug 2020
5. Overview | Coronavirus Disease COVID-19 (n.d.), https://www.COVID19treatmentguidelines.nih.gov/overview/. Accessed 28 Aug 2020
6. COVID-19 basics - Harvard Health (n.d.). https://www.health.harvard.edu/diseases-and-conditions/COVID-19-basics. Accessed 28 Aug 2020
7. Coronavirus & COVID-19 Overview: Symptoms, Risks, Prevention, Treatment & More. (n.d.), https://www.webmd.com/lung/coronavirus. Accessed 28 Aug 2020
8. T. Ozturk, M. Talo, E.A. Yildirim, U.B. Baloglu, U. Ozal Yildirim, R. Acharya, Automated detection of COVID-19 cases using deep neural networks with X-ray images. Comput. Biol. Med. **121**, 103792 (2020)
9. J.P. Cohen, COVID-19 image data collection (2020), https://github.com/ieee8023/COVID-chestxray-dataset
10. T. Mahmud, M.A. Rahman, S.A. Fattah, CovXNet: a multi-dilation convolutional neural network for automatic COVID-19 and other pneumonia detection from chest X-ray images with transferable multi-receptive feature optimization. Comput. Biol. Med. **122**, 103869 (2020)
11. D. Kermany, K. Zhang, M. Goldbaum, Labeled optical coherence tomography (OCT) and chest X-ray images for classification, version 2 (Mendeley Data, 2018)
12. S. Fujieda, K. Takayama, T. Hachisuka, Wavelet convolutional neural networks, arXiv preprint arXiv:1805.08620 (2018)
13. H. Panwar, P.K. Gupta, M.K. Siddiqui, R. Morales-Menendez, V. Singh, Application of deep learning for fast detection of COVID-19 in X-rays using nCOVnet. Chaos Solitons Fractals **138**, 109944 (2020)
14. A.I. Khan, J.L. Shah, M.M. Bhat, Coronet: a deep neural network for detection and diagnosis of COVID-19 from chest x-ray images. Comput. Meth. Programs Biomed. **196**, 105581 (2020)
15. F. Ucar, D. Korkmaz, COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease 2019 (COVID-19) from X-ray images. Med. Hypotheses **140**, 109761 (2020)
16. L. Wang, A. Wong, COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-Ray images, arXiv preprint arXiv:2003.09871 (2020)
17. N.N. Das, N. Kumar, M. Kaur, V. Kumar, D. Singh, Automated deep transfer learning-based approach for detection of COVID-19 infection in chest X-rays. IRBM (2020)
18. S. Toraman, T.B. Alakuş, İ Türkoğlu, Convolutional capsnet: a novel artificial neural network approach to detect COVID-19 disease from X-ray images using capsule networks. Chaos Solitons Fractals **140**, 110122 (2020)
19. Capsule neural network - Wikipedia (n.d.), https://en.wikipedia.org/wiki/Capsule_neural_network. Accessed 6 Sept 2020
20. M. Rahimzadeh, A. Attar, 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. Inf. Med. Unlocked **19**, 100360 (2020)
21. J. Shuja, E. Alanazi, W. Alasmary, A. Alashaikh COVID-19 datasets: a survey and future challenges (medRxiv 2020)
22. Radiologists find chest X-rays highly predictive of COVID-19 (n.d.) https://medicalxpress.com/news/2020-09-radiologists-chest-x-rays-highly-COVID-.html. Accessed 7 Sept 2020
23. COVID-19 | Radiology Reference Article | Radiopaedia.org. (n.d.). https://radiopaedia.org/articles/COVID-19-4?lang=us. Accessed 7 Sept 2020
24. K. Krishnamurthy, Spectral Image Processing Theory and Methods: Reconstruction, Target Detection, and Fundamental Performance Bounds. Dissertation, Duke University (2011). https://hdl.handle.net/10161/3945

25. P. Parida, N. Bhoi, Wavelet based transition region extraction for image segmentation. Fut. Comput. Inf. J. **2**(2), 65–78 (2017)
26. W. El-Shafai, F.A. El-Samie, Extensive COVID-19 X-ray and CT chest images dataset, version 3 (Mendeley Data 2020). https://doi.org/10.17632/8h65ywd2jr.3
27. COVID-Net/COVIDx.md at master, lindawangg/COVID-Net, GitHub (n.d.), https://github.com/lindawangg/COVID-Net/blob/master/docs/COVIDx.md. Accessed 24 Dec 2020
28. COVID-19 Radiography Database | Kaggle. (n.d.), https://www.kaggle.com/tawsifurrahman/covid19-radiography-database. Accessed 24 Dec 2020
29. COVID-19 DATABASE | SIRM. (n.d.). https://www.sirm.org/en/category/articles/covid-19-database/. Accessed 24 Dec 2020