Classification of Chest X-Ray images to distinguish between COVID-19 and Pneumonia
COMP 6721 Applied Artificial Intelligence
Final Report - By Group D, Concordia University

Anonymous CVPR submission

Paper ID *****

Abstract

A virus, COVID-19 has spread across the world as a pandemic, severely affecting people's health and well being. A systematic methodology for detecting COVID-19 can be created by integrating radiological analysis with machine learning. The most practical solution is chest X-ray imaging. One of the most effective ways to combat COVID-19 is through early diagnosis using these chest X-ray images. In order to distinguish Covid from other lung diseases, the examined works in this paper train CNN models AlexNet, VGG11, and ResNet50 using AI with image processing of unprocessed chest X-rays and CT scans. It is possible to see how transfer learning is used. On three datasets, the properties of which are also presented, the models are trained and validated. These algorithms' outputs are compared subsequently using performance metrics like accuracy, precision, recall, and F1 score. The limited availability of COVID image data and the high accuracy of the prediction of the severity of patients using deep learning compared to well-known methods of COVID-19 detection like PCR tests, which are reliable and are preferred because of their low cost, availability, and simplicity, are major challenges faced in this research domain.

1. Introduction

COVID-19 is an infectious illness brought on by the coronavirus strain SARS-CoV-2 responsible for the severe acute respiratory syndrome. Patients suffering from COVID-19 can also present with abnormalities on chest X-ray images that are characteristic of infection [1]. According to WHO, the most common diagnosis in severe COVID-19 patients is severe pneumonia [2]. Due of the high rate of infection in China and other nations within a short period of time, the WHO (World Health Organization) designated it a pandemic on February 11, 2020. [3, 4] Testing with RT-PCR is the most popular diagnosing technique. Due to their limited sensitivity, they also have a significant rate of false-negative and false-positive results [5]. Consequently, chest radiography, which includes chest X-rays (CXRs) and chest tomography (CT) scans, can be another option for the early identification of COVID-19. In order to process CXR and CT scan images and produce accurate COVID-19 detection results, an automated diagnosis system is required.

With people all round the world suffering from COVID, it is important to diagnose it correctly by identifying it from numerous similar cases of pneumonia, Tuberculosis, etc. [2]. Hence, it is our model topic to distinguish between the chest X-ray images of COVID and other diseases related to chest as it can be helpful in the early diagnosis of COVID-19.

With our proposed ML and CNN based models-AlexNet, VGG11 and ResNet50, we aim to differentiate the X-ray images into Normal, COVID-19, Pneumonia and few other classes based on 3 datasets with over 22000 images. Each dataset having multiclasses.

With huge amount of data, during the implementation of the project, we faced challenges with respect to time as our models were taking enormously large time to learn. We have used multiple resources. We have tried multiple sorts of GPUs, CPUs, and Online GPUs. Initially, for the implementation of our models, we worked on local CPUs and internal GPUs (eg. Intel UHD graphics) which took a huge amount of time. Approximately 6000 sec- onds to complete the 20 epochs. After that, we set up our project models on Google Collab and by uploading datasets to Google Drive. We performed the training model with 5 epochs in Google Colab and the results were satisfying. Af- ter a while, we shifted our project from Google Colab to Kaggle GPU as google collab was not performing well for 50 epochs. Also, we used NVIDIA GPU, as both were performing quite similarly and training time for 50 epochs was around 1 hour. Finally, our final model training was trained on Kaggle GPU, Google Colab, and NVIDIA GPU.

Another challenge was that the dataset consists of images of different sizes, for which we used the pre-processing of data to get all images to uniform size.

Also, there was unlabelled data in our dataset 3, where labels of images are in a separate csv file not labelled to image dataset, which was a challenge which we addressed by using tensorflow keras preprocessing method.

The goal throughout developing the application is to study the impact of different training models in our application by interchanging the datasets, CNN architectures, hyper parameters etc. which can be seen in our results mentioned further using the metrics like Accuracy, Precision, Recall and F1 scores.

1.1. Related works

Given that the conventional PCR process takes six to eight hours to complete, to identify COVID-19 infection, some recent studies have suggested using image classifiers solutions have been designed to give medical professionals another quick and affordable method to recognize pneumonia infections like COVID-19. A procedure used on people with symptoms is CT scanning. The sensitivity of RT-PCR is less than that of CT, according to [14]. As a result, CT scanning is more accurate than X-ray imaging, even if it is more expensive, and it sometimes helps by giving more information.

The authors of [16] provide five transfer learning models for COVID-19 detection in lung CT images. The research evaluates the application of conventional and contrasting adaptive histogram equalization in lung scans. The effectiveness and effects of utilizing histogram equalization methods on various learning models are not, however, demonstrated in this work. The authors of [17] create a dataset of X-ray images and propose a semi-automated pre-processing methodology to pre-train deep learning models to recognize COVID-19 and other diseases with well-known features. The model employed enables the reduction of X-ray image noise. According to the experimental tests, even straightforward network models like the VGG19 gain accuracy (by 83%).

The importance of using AI techniques for image analysis for the detection and management of COVID-19 cases has been described by researchers in [20]. By analyzing pulmonary CT data with deep learning models, COVID-19 may be detected with accuracy [20]. An open-source COVID-19 diagnosis system based on a deep CNN was created by researchers in [21]. The detection of COVID-19 patients using X-ray images has been reported in this study utilizing a customized deep CNN configuration. Another important work has discussed the X-ray dataset, which includes X-ray images from people with common pneumonia, COVID-19 patients, and healthy individuals [22]. The latest CNN architectures are used in this study to identify COVID-19 patients automatically. Transfer learning's COVID-19 detection accuracy of 97.82% is encouraging in this study. Another recent study that is pertinent examines the validity and applicability of deep CNNs of the Decompose-, Transfer-, and Compose types for the detection of COVID-19 using chest X-ray image classification [23]. According to the study's authors, the accuracy, sensitivity, and specificity of the method were 95.12%, 97.91%, and 91.87%, respectively.

High topological knowledge, in-depth parametric information, and a sizable dataset are necessary for the usage of deep learning. As a result, it is challenging for the region with fewer skilled labor to embrace CNN architecture. In this study, we concentrated on creating a simplified computerized methodology employing a feature-based conventional classification strategy with the highest level of accuracy for categorizing X-ray images with the goal of detecting COVID-19.

2. Methodology

There are prominent COVID-19-related features in the chest X-ray image. The detection of the disease will be aided by the computerised extraction of such traits. In this study, we seek to create a more straightforward method for detecting abnormalities in chest X-ray images.Convolution layers and other pooling layers make up CNN architectures. They excel in computer vision classification tasks, which makes it possible for them to evaluate medical images. The subsequent subsections examine different CNN architectures and their methodologies for identifying COVID-19 patients from unprocessed X-ray images.

2.1. Datasets

Dataset 1:
"COVID-19 Radiography Database" [8]
This dataset consists of 21,164 images which are categorized into 4 classes [COVID Positive, Normal, Non-COVID lung infection, Pneumonia]. Their number of images per class is as follows: COVID-19: 3616 Images, Normal: 10,192 Images, Lung Opacity: 6012 Images (This class represents the infection of lungs which is not caused by Covid-19) Viral Pneumonia: 1345 Images. All the images are in the Portable The network Graphics (PNG) file

CVPR #*****

format and shape of the image are 299x299px. This data is collected from Germany medical schools, GitHub, Kaggle, Radiological Society of North America(RSNA) and various other public datasets. The preview of the images of each class from Dataset1 is as shown in the below image.


Class: Covid

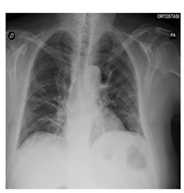

Class: Covid


Class: Lung Opacity

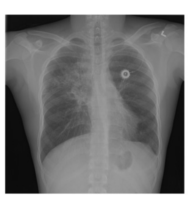

Class: Pneumonia

Dataset 2 :
"COVID-19, Pneumonia, Normal Chest Xray Dataset" [9]
This dataset has 3 classes containing 6939 multi-type images. The number of images per class is 2313 Images. Each class contains a file format of PNG for all images and the average shape of images in all classes is 1024x1024px. We have collected this data from Kaggle but it was originally gathered using GitHub, the Radiopaedia, the Italian Society of Radiology (SIRM), and Figshare data repository websites. Then,912 augmented images were collected from Mendeley.


Class: Covid


Class: Covid


Class: Normal


Class: Pneumonia

Dataset 3 :
"Xray Body Parts-fastai" [10]
This particular dataset contains the largest number of classes compared to our other image dataset. The number of Classes is 22. This dataset contains a total of 2,482 images. And Images per class are Train class: 1,738 images Test class: 743. Among the dataset the relevant class type to our project was class Chest, this class has 724 Images from the train and 49 for the test. Other 1,014 images are distributed among the rest of the 21 classes. The image type for this dataset is PNG, and all images' shape is 420x512 px. The average number of images per class is less than 49 images. Some of the types have very few images. Forex Clavicles class there are only 9 images only.


Class: Covid


Class: Covid

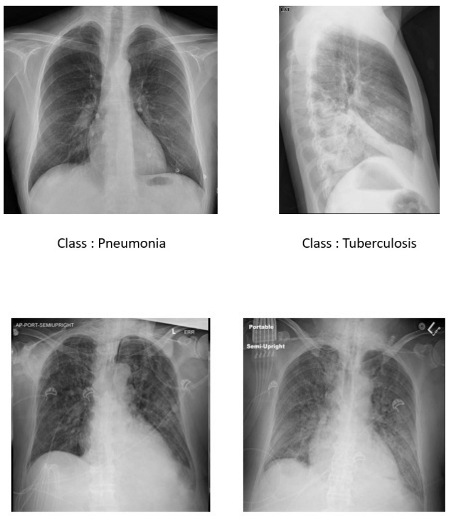

Class: Wrist


Class: Cervical Spine

There is unlabelled data in dataset 3, and also data which is irrelevant to our project, hence while using dataset 3 with the existing CNN models, it was not working out, and at the moment we had to find a new dataset which need to have similar or more number of classes as per our previous dataset 3. Luckily we found a github repository named: covid-chesxray-dataset. This dataset has images of 27 classes and also it was containing "data-loaders" which we have used in our earlier successful training of model for dataset1 and dataset2. So we finalized the dataset and we trained our models with this dataset afterwards.

Updated Dataset 3 :
"covid-chestxray-dataset" [11]
It is COVID-19 image data collection which is built by publicly released images as well as through indirect collections from hospital and physicians. All the images exist in the dataset is also approved by University of Montreal's Ethics Committee. The number of classes in this dataset is 27 datasets. It consists of various other classes of diseases such as Muycoplasma, Pneumonia, SARS, Varicella, Streptococ-

cus, etc. This dataset has a total 654 images which are distributed as : 481 images for training and 173 images for testing. After working with this dataset 3 we got to know that there exists some extra labels which are there but not relavent for the application. Example of the classses were ”unknown” and ”todo”. We also go to know about more classes while pre-processing the data. In a nutshell there are 19 main classes of diseases and extra 8 classes were also there.



Class : Pneumonia
Class : Tuberculosis
Class : Covid-19
Class : Tuberculosis

2.2. CNN Models

For the implementation of the model, we have selected 3 CNN architectures are VGG11, ResNet50 and AlexNet.

AlexNet:
In comparison of some trained transfer learning models, this architecture takes less training time and fewer eras. It also produces excellent outcomes when it comes to the identification and categorization of photos. On the ImageNet dataset, this network is also renowned for providing the greatest accuracy. The network has a depth size of 8 and has 5 convolutions, 2 hidden, and 1 fully linked layer. The input image was 227 by 227 and had 61 million fine-tuned characteristics. Overfitting was addressed using the dropout method, which also allowed the network to pick up additional features. The activation function of ReLU was employed. It used the nonsaturating ReLU activation function, which showed improved training performance over tanh and sigmoid. [8]

In our project, we used a vanilla version of AlexNet. According to our use case which distinguishes between COVID-19 and Pneumonia, networks with less number of layers would not find many features so we decided to use AlexNet as our first approach in Convolution Neural Networks.

VGG11:
The Visual Geometry Group proposed this neural network, which demonstrated excellent performance in the 2014 ImagNet Large Scale Visual Recognition Challenge (ILSVRC) (VGG). It can achieve good accuracy and typically consists of 16 or 19 convolution layers. AlexNet is primarily capable of object detection. It takes into account over-fitting by using data augmentation and dropout. VGG replaces the tanh activation function with ReLU by encapsulating its distinct features for over-pooling. VGG came into the picture as it addresses the depth of CNN.Input. VGG takes in a 224x224 pixel RGB image. The convolutional layers in VGG use a tiny receptive field. There are also 1x1 convolution filters which act as a linear transformation of the input, which is followed by a ReLU unit. VGG has three fully-connected layers: the first two have 4096 channels each and the third has 1000 channels, 1 for each class. All of VGG’s hidden layers use ReLU.

To go to the next step we decided to go to VGG has all the capabilities of AlexNet and it goes next step and tries to get a better result, via the ReLU activation function and other changes in its layers. the main reason that we used VGG11 is that, it is the easiest to implement and will form the basis for other configurations and training for other VGG models as well.

ResNet50
The training error rises as the network depth in CNNs increases. This issue is resolved by ResNet’s use of residual units. Two deep layers are present in ResNet-18 and ResNet-34, whereas three deep layers are present in ResNet-50/101/152. The layer that does not contribute to the solution is skipped by the residual learning component, which reuses the activation from earlier levels. It reduces the vanishing gradient issue and boosts speed by using batch normalisation and identity connection. One of the CNNs that is most frequently employed in COVID diagnosis research, according to observations, is ResNet [24]. ResNet is a gateless or open-gated variant of the HighwayNet,[2] the first working very deep feedforward neural network with hundreds of layers, much deeper than previous neural networks. Typical ResNet models are implemented with double- or triple-layer skips that contain nonlinearities (ReLU) and batch normalization in between. Models with several parallel skips are referred to as DenseNets. ResNet50 is a variant of the ResNet model which has 48 Convolution layers along with 1 MaxPool and 1 Average Pool layer. It is a widely used ResNet model.

The reason that we used ResNet is that ResNets is one of the most efficient Neural Network Architectures, as they help in maintaining a low error rate much deeper in the network. Moreover, since we wanted to use the basic version of ResNets, we choose ResNet50, due to several reasons, first is the most popular ResNets, and also it has 50 convolutional layers which could get deeper features from the photo and since our photos are X-Ray we need this kind of layers.

2.3. Optimization algorithm

Adam was presented by Diederik Kingma from OpenAI and Jimmy Ba from the University of Toronto in their 2015 ICLR paper (poster) titled “Adam: A Method for Stochastic Optimization“. [15] It is an adaptive learning rate optimization algorithm. The name “Adam” derives from the phrase “adaptive moments.” In the context of the earlier algorithms, it is perhaps best seen as a variant of the combination of RMSProp and momentum with a few important distinctions. First, in Adam, momentum is incorporated directly as an estimate of the first-order moment (with exponential weighting) of the gradient. The most straightforward way to add momentum to RMSProp is to apply momentum to the rescaled gradients. The use of momentum in combination with rescaling does not have a clear theoretical motivation. Second, Adam includes bias corrections to the estimates of both the first-order moments (the momentum term) and the (uncentered) second-order moments to account for their initialization at the origin.

For all 11 models, we used accuracy precision, recall, f1 score, and confusion matrix. We used 3 datasets with 3 different models along with 2 other pre-trained VGGs for datasets 1 and 2. We used 50 epochs and a learning rate of 0.0001, CrossEntropyLoss for loss function, and a batch size of 32. We also tested with other learning rates to measure its impact on our models.

For the third dataset, we were forced to use other loss functions since the third dataset has multi-label data, and also there are 27 classes. We have used BCEWithLogitsLoss as the loss function.

3. Results

3.1. Experiment Setup

Parameters and Hyperparameters: For all the models, we used a batch size of 32, Cross Entropy Loss as a loss function, and a learning rate of 0.0001 and 50 epochs.

Attempts: We tried 3 models AlexNet, VGG11 and ResNet50. Accuracy, F1 Score, Recall, and precision for each model are available below.

Preprocessing: We used tranforms to perform preprocessing of data. horizontal flip, central crop, normalize image with mean, resizing images to 224*224 is all done as part of the preprocessing.

Accuracy			
DS/Model	AlexNet	VGG11	ResNel50
Dataset1	91.51	94.36	90.94
Dataset2	93.12	95.48	94.26
Dataset3	77.54	75.03	64.02

Precision			
DS/Model	AlexNet	VGG11	ResNel50
Dataset1	0.9361	0.9461	0.9090
Dataset2	0.9351	0.9568	0.9451
Dataset3	0.2143	0.0815	0.1945

Recall			
DS/Model	AlexNet	VGG11	ResNel50
Dataset1	0.9106	0.9550	0.9236
Dataset2	0.9317	0.9549	0.9429
Dataset3	0.1148	0.0492	0.1115

F1 Score			
DS/Model	AlexNet	VGG11	ResNel50
Dataset1	0.9152	0.8880	0.9095
Dataset2	0.9313	0.9548	0.9426
Dataset3	0.2692	0.1154	0.2615

The rate or speed at which the model learns is controlled by the learning rate hyperparameter. In particular, it regulates how much allocated error is added to the model’s weights each time they are updated, such as at the conclusion of each batch of training instances. We used a lr of 0.0001

Increasing epochs to 50 because there was a lot of data in our dataset. However, our model eventually reached a point where increasing epochs will not improve accuracy. At this point, we considered playing around with our model’s learning rate. Changed lr from 0.01, 0.1 to 0.0001.

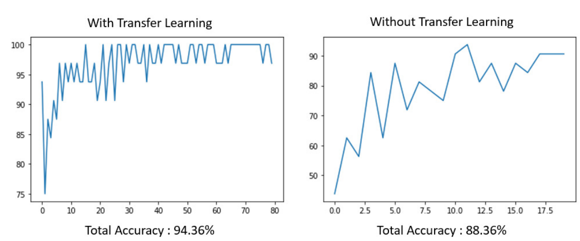
One of the most important hyper parameters in machine learning is batch size. How many samples must be processed before the internal model parameters are updated is determined by the batchsize. It might be among the most crucial steps you do to make sure your models work as effectively as possible. We have chosen our batch size as 32.

What is and isn’t a good forecast is determined by loss functions. In other words, the effectiveness of our estimator depends on the loss function we choose. We convert the learning problem into an optimization problem, define a loss function and then optimize the algorithm to minimize the loss function.We have used Cross Entropy loss as our loss function.

3.2. Main Results

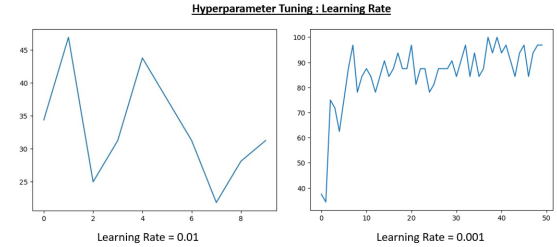
Based on the metrics above, we can see that our models worked very well on first and second dataset which has 3 and 4 classes. For dataset 3 there is sort of an issue. Firstly, there are 27 classes and each image has multi label. It was making the training harder. As we train our models there was not much progress after couple of epochs, the reason could be several things. Firstly, we got stuck in local minima. Secondly, since it has multi-label it cannot perform better. Thirdly, our networks is optimized for only particular label and class. Lastly getting about 70% accuracy for third dataset is enough since it has 27 classes.

Comparison of CNN models:
All the CNN architectures almost had same time of training on google colab GPUs except VGG. VGG took more time since it has more layers in its architectures. Other beautiful thing we observed is that, as VGG is the next generation for the AlexNet, its results are better than AlexNet. Moreover,after we tried to use pre-trained VGG, the result was as we expected. The training will reach to a perfect accuracy sooner.



Results of Dataset 1

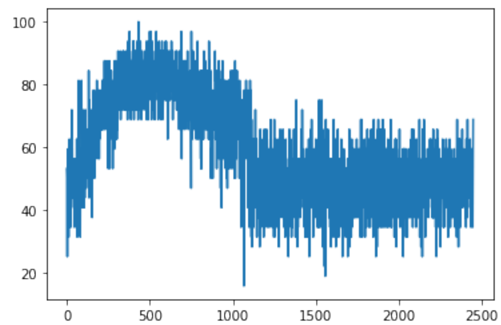
Hyperparameter Tuning : Learning Rate



The above image vividly shows the difference between pre-trained and usual models. Pre-trained model get 94.36% accuracy after only 20 epochs for dataset1 and for dataset 2 it get 95.48% accuracy.

Failed efforts:

First and obvious failure was to use 0.01 and 0.1 learning rate which made our models stuck in local minima. Other failed attempts is to use SGD with lr=0.0001, momentum=0.9, weight_decay=0.5. For this scenario the result was really bad and the plot below shows why:



As per the plot, optimizer has put net into local extremum.

3.3. Ablative Study

In order to see the impact of changing hyperparameter on the models, we changed learning rate and observed how traning works. Using 0.01 as lr showed that the training will not stuck in proper place, which means that this learning rate make model steps longer. Moreover, this long steps will avoid model to train properly. hence we have used lr = 0.0001.

We also changed the batch size in order to see its impact, and impact was as we expected. Reducing batch size caused longer training time but not necessarily gave us better performance. On the other hand, increasing batch size made training harder since we cannot have proper tuning. So 32 was our sweet spot for the batch size which gave us acceptable result.

It is observed in our models that, the number of classes is inversely proportional to the accuracy. As the number of classes in a dataset increase, the feature extraction becomes complex and results in relatively less accuracy when compared to a model trained on a dataset with less number of classes. We observed that the accuracy of dataset 1 and dataset 2 on the models AlexNet, VGG11 and ResNet50 are almost of similar accuracies 90%. These datasets 1 and 2 have classes 4 and 3 respectively. Meanwhile, dataset 3 having the largest number of classes, 27, is observed to produce less accuracy in the range of (65%, 78%) for the three CNN models. Not only the accuracy, but we can also observe how number of classes is affecting precision, recall and F1-score in our results above.

Number of images in the dataset is directly proportional to the accuracy, precision, recall and F1-score. In our project, Dataset 1 with 22000 images is observed to have the highest of metrics, while dataset 2 with 7000 images is moderate and dataset 3 is relatively very less.

In conclusion, we have learned the architecture of CNN and it's applications althrough the project. It was a great experience to practise image classification from scratch and make utilization of our time effectively.

4. References

References

[1] Ng Ming-Yen, et al. *Imaging profile of the COVID-19 infection: radiologic findings and literature review* Radiology: Cardiothoracic Imaging 2.1 (2020). 1

[2] *https://www.who.int/docs/defaultsource/coronavirus /clinical-management-of-novelcov.pdf* 1

[3] D. Cucinotta and M. Vanelli, *"WHO declares COVID-19 a pandemic,"* Acta Biomedica: Atenei Parmensis, vol. 91, pp. 157–160, 2020. 1

[4] F. Rustam, A. A. Reshi, A. Mehmood et al., *"COVID-19 future forecasting using supervised machine learning models,"* IEEE Access, 2020. 1

[5] Chan J.-W., Yuan S., Kok K.-H., To K.-W., Chu H., Yang J., Xing F., Liu J., Yip C.-Y., Poon R.-S., Tsoi H.-W., Lo S.-F., Chan K.-H., Poon V.-M., Chan W.-M., Ip J.D., Cai J.-P., Cheng V.-C., Chen H., Hui C.-M., Yuen K.-Y. *A familial cluster of pneumonia associated with the 2019 novel coronavirus indicating person-to-person transmission: a study of a family cluster.* Lancet. 2020;395(10223):514–523. [PMC free article] [PubMed] [Google Scholar] [Ref list 1

[6] M.E.H. Chowdhury, T. Rahman, A. Khandakar, R.Mazhar, M.A. Kadir, Z.B. Mahbub, K.R. Islam, M.S.Khan, A. Iqbal, N. Al-Emadi, M.B.I. Reaz, M. T. Islam *"Can AI help in screening Viral and COVID-19 pneumonia?"* IEEE Access, Vol. 8, 2020, pp. 132665 - 132676.

[7] Rahman, T., Khandakar, A., Qiblawey, Y., Tahir, A.,Kiranyaz, S., Kashem, S.B.A., Islam, M.T., Maadeed, S.A., Zughaier, S.M., Khan, M.S. and Chowdhury *Exploring the Effect of Image Enhancement Techniques on COVID-19 Detection using Chest X-ray Images* arXiv preprint arXiv:2012.02238, M.E., 2020.

[8] Dataset 1: *https://www.kaggle.com/datasets/tawsifurrah man/covid19-radiography-database* 2, 4

[9] Dataset 2: *https://www.kaggle.com/datasets/amanullah asraf/covid19-pneumonia-normal-chest-xray-pa-dataset* 3

[10] Dataset 3: *https://www.kaggle.com/code/alexanderyyy /xraybodypartsfastai/notebook* 3

[11] Updated Dataset 3: *https://github.com/ieee8023 /covid-chestxray-dataset* 3

[12] *https://github.com/agchung*

[13] *https://figshare.com/articles/dataset/COVID19ChestX-RayImageRepository/12580328/2*

[14] Santura, I.; Kawalec, P.; Furman, M.; Bochenek, T. *Chest computed tomography versus RT-PCR in early diagnostics of COVID-19—A systematic review with meta-analysis.* 2021. 2

[15] *https://doi.org/10.48550/arXiv.1412.6980* 5

[16] Lawton, S.; Viriri, S. *Detection of COVID-19 from CT Lung Scans Using Transfer Learning.* Comput. Intell. Neurosci. 2021, Diagnostics 2022. 2

[17] Horry, M.J.; Chakraborty, S.; Paul, M.; Ulhaq, A.; Pradhan, B.; Saha, M.; Shukla, N. *X-ray Image Based COVID-19 Detection Using Pre-trained Deep Learning Models.* engrXiv 2020. 2

[18] Padma, T.; Kumari, C.U. *Deep Learning Based Chest X-ray Image as a Diagnostic Tool for COVID-19.* In Proceedings of the 2020 International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 10–12 September 2020; pp. 589–592.

[19] Karhan, Z.; Akal, F. *Covid-19 Classification Using Deep Learning in Chest X-ray Images.* In Proceedings of the 2020 Medical Technologies Congress (TIPTEKNO), Antalya, Turkey, 19–20 November 2020; pp. 1–4.

[20] D. Dong, Z. Tang, S. Wang et al., *The role of imaging in the detection and management of COVID-19.* IEEE Reviews in Biomedical Engineering, vol. 14, pp. 16–19, 2020. 2

[21] L. Wang and A. Wong, *COVID-Net: A tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images.* 2020, http://arxiv.org/abs/ 2003.09871. 2

[22] I. D. Apostolopoulos and T. A. Mpesiana, *Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks.* Physical and Engineering Sciences in Medicine, vol. 43, pp. 635–640, 2020. 2

[23] A. Abbas, M. M. Abdelsamea, and M. M. Gaber *Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network.* 2020, http://arxiv.org/abs/ 2003.13815. 2

[24] He, K.; Zhang, X.; Ren, S.; Sun, J. *Deep residual learning for image recognition.* In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. 4