

# COVID-19 detection from chest X-Ray images using Deep Learning and Convolutional Neural Networks

Antonios Makris  
Department of Informatics and  
Telematics, Harokopio University of  
Athens  
Athens, Greece  
amakris@hua.gr

Ioannis Kontopoulos  
Department of Informatics and  
Telematics, Harokopio University of  
Athens  
Athens, Greece  
kontopoulos@hua.gr

Konstantinos Tserpes  
Department of Informatics and  
Telematics, Harokopio University of  
Athens  
Athens, Greece  
tserpes@hua.gr

## ABSTRACT

The COVID-19 pandemic in 2020 has highlighted the need to pull all available resources towards the mitigation of the devastating effects of such "Black Swan" events. Towards that end, we investigated the option to employ technology in order to assist the diagnosis of patients infected by the virus. As such, several state-of-the-art pre-trained convolutional neural networks were evaluated as of their ability to detect infected patients from chest X-Ray images. A dataset was created as a mix of publicly available X-ray images from patients with confirmed COVID-19 disease, common bacterial pneumonia and healthy individuals. To mitigate the small number of samples, we employed transfer learning, which transfers knowledge extracted by pre-trained models to the model to be trained. The experimental results demonstrate that the classification performance can reach an accuracy of 95% for the best two models.

## CCS CONCEPTS

• **Computing methodologies** → **Neural networks; Artificial intelligence**; • **Applied computing** → *Health informatics; Life and medical sciences.*

## KEYWORDS

Covid-19, SARS-CoV-2, Chest X-ray, Deep Learning, Convolutional Neural Networks, Transfer Learning, Medical Image Classification

### ACM Reference Format:

Antonios Makris, Ioannis Kontopoulos, and Konstantinos Tserpes. 2020. COVID-19 detection from chest X-Ray images using Deep Learning and Convolutional Neural Networks. In *11th Hellenic Conference on Artificial Intelligence (SETN 2020)*, September 2–4, 2020, Athens, Greece. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/3411408.3411416>

## 1 INTRODUCTION

The year 2020 has been marked by the pandemic disease caused by a type of the corona virus family (CoV), called COVID-19 or SARS-CoV-2, which has led to over four million infections and more than

355,000 deaths worldwide. COVID-19 is a Severe Acute Respiratory Syndrome (SARS) that was firstly identified in Wuhan, China in December 2019 and has rapidly spread globally in a few months, making it a highly contagious virus. The virus is characterized by symptoms that mostly relate to the respiratory system and include shortness of breath, loss of smell and taste, cough and fever, a range of symptoms that is shared among other types of viruses such as the common cold.

Unlike other viruses, COVID-19 has a long incubation period, ranging from 3 days to 13 days, although on average, the time from exposure to symptom onset is approximately five to six days. The long incubation period makes COVID-19 more contagious since people carrying the virus will most probably keep interacting with other people until they realize they have the virus, leading to more infections. Furthermore, it has been reported that several patients carrying the disease might not show any symptoms at all (asymptomatic patients). The combination of both the long incubation period and the asymptomatic cases, makes the corona virus even harder to detect, trace and contain, which explains its rapid spread.

Several studies have been conducted since the beginning of the year 2020 that try to develop a methodology to identify patients carrying the disease [1, 2, 6]. The majority of these studies in the field of computer science, employ convolutional neural networks (CNNs) in order to classify images of CT scans or X-Rays of the chest as normal or not, in an attempt to identify possible cases of the corona virus. The widespread usage of CNNs for image classification tasks is due to the fact that they have demonstrated a high-accuracy performance in the fields of image recognition and object detection [17]. Their success lies in the fact that they are able to capture hidden features of the images, through their numerous hidden layers.

In this research work the main contributions are:

- The effectiveness of several state-of-the-art pre-trained CNNs was evaluated regarding the automatic detection of COVID-19 disease from chest X-Ray images. Transfer learning was employed in order to obtain high prediction accuracy from a small sample of X-ray images.
- A large number of CNN architectures were employed in an attempt to not only distinguish X-Rays between COVID-19 patients and people without the disease, but to also discriminate pneumonia patients from patients with the corona virus, acting as a classifier of respiratory diseases. Note that the developed methodology cannot replace direct clinical diagnoses.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](https://permissions.acm.org).

SETN 2020, September 2–4, 2020, Athens, Greece

© 2020 Association for Computing Machinery.

ACM ISBN 978-1-4503-8878-8/20/09...\$15.00

<https://doi.org/10.1145/3411408.3411416>

The rest of the paper is structured as follows. Section 2 presents a two-fold literature review: i) the usage of deep learning for image classification and ii) the usage of deep learning for the detection of COVID-19. Section 3 describes the methodology employed towards the identification of the corona virus through X-Ray scans, while Section 4 presents the research findings and the experimental results. Finally, Section 5 concludes the merits of our work and presents roadmaps for future research.

## 2 RELATED WORK

### 2.1 Deep learning approaches for image classification and object detection

Numerous studies have used Convolutional Neural Networks for the problem of image classification in the literature, most of which create different architectures for the neural networks. Deep convolutional neural networks are one of the powerful deep learning architectures and have been widely applied in a broad range of machine learning tasks. The first CNN to create a standard “architectural template” was the LeNet-5 [12], which uses two convolutional layers and three fully-connected ones. Ever since, more architectures followed that used the same idea of adding more convolutions and pooling layers, ending with one or more fully-connected ones. Before making more variations on the architectures, researchers continued using more layers and creating deeper networks and as a result, VGG-16 [19] was emerged. VGG-16 used 13 convolutional layers and 3 fully connected ones while VGG-19, a successor of the previous network, simply added more layers.

The years that followed, researchers, apart from making the networks deeper, added more complexity by introducing several techniques inside the layers of the networks. Inception-v1 [22] besides the fact that it uses 22 layers in total, it also uses a “network inside a network” approach by using “Inception” modules. Sharing a similar architecture, Inception-v3 [21], a successor of the previous network, was among the first networks to use batch normalization to the layers. The same authors of the previous networks introduced a family of a new architecture, called Inception-ResNet-v2 [20], in which they converted the Inception modules to Residual Inception blocks, created a new type of Inception modules and added more of these to the network, making it even deeper. ResNet-50 [7] was also among the first networks to use batch normalization. Moreover, it had an even deeper architecture (152 layers) and it used skip connections or residuals. Xception [3] replaced the Inception modules with depthwise separable convolutions. Similarly to Xception, MobileNetV2 [18] uses depthwise separable convolutions, which reduce the complexity and size of the network. Finally, the “battle” for a better network architecture continued and resulted in several other CNNs, each one introducing a different modification, such as DenseNet [10], NASNet [23] and ResNet152V2 [15].

### 2.2 Deep learning approaches for COVID-19 detection based on image classification

Various research studies already exist for COVID-19 detection. For the most part, deep learning techniques are employed on chest radiography images with a view to detect infected patients and the results have been shown to be quite promising in terms of accuracy.

In [14] a deep convolutional neural network able to predict the coronavirus disease from chest X-ray (CXR) images is presented. The proposed CNN is based on pre-trained transfer models in order to obtain high prediction accuracy from a small sample of X-ray images. The images are classified into two classes, normal and COVID-19. To overcome the insufficient data and training time, a transfer learning technique is applied by employing the ImageNet dataset. The results showed the superiority of ResNet50 model in terms of accuracy in both training and testing stage. Abbas et al. [1] presented a novel CNN architecture based on transfer learning and class decomposition in order to improve the performance of pre-trained models on the classification of X-ray images. The ResNet18 pre-trained ImageNet network is used and the results showed an accuracy of 95.12% on CXR images. Farooq et al. [6] presents a CNN framework for COVID-19 detection from other pneumonia cases which utilizes a three step technique to fine-tune a pre-trained ResNet-50 architecture in order to improve performance and reduce training time. Progressive resizing of input images and fine-tuning of network at each stage manages to achieve a better generalization and an increased performance (96.23% accuracy). Hemdan et al. [8] presented a framework consist of seven deep learning image classifiers called COVIDX-Net with a view of classifying COVID-19 disease from CXR images. As the results showed, the best performance achieved for the VGG19 and DenseNet201 classifiers with an accuracy of 90%. Finally, Apostolopoulos et al. [2] evaluated the performance of five pre-trained CNN networks regarding the detection of COVID-19 from CXR. The results showed that VGG19 and MobileNetv2 achieved the higher accuracy, 93.48% and 92.85% respectively.

## 3 METHODOLOGY

### 3.1 Dataset Description

The dataset used in this study contains chest X-Ray images from patients with confirmed COVID-19 disease, common bacterial pneumonia and normal incidents (no infections) and is a combination of two different publicly available datasets. More specifically, COVID-19 cases have been obtained from Dr. Joseph Cohen’s Github repository [4] and consist of 112 Posterior-Anterior (PA) X-ray images of lungs. In general, this repository contains chest X-ray / CT images of patients with acute respiratory distress syndrome (ARDS), COVID-19, Middle East respiratory syndrome (MERS), pneumonia and severe acute respiratory syndrome (SARS). In addition, 112 normal and 112 pneumonia (bacterial) chest X-Ray images were selected from Kaggle’s repository<sup>1</sup>. In summary, the dataset used for this work is evenly distributed regarding the number of cases and consist of 3 classes (covid, pneumonia and normal) and it is publicly available in<sup>2</sup>.

The reason that we are using such a relatively small dataset is mainly due to the small number of confirmed COVID-19 samples. In order to keep the dataset balanced, we selected only the same number of samples for the other classes. Furthermore, to the best of our knowledge the pneumonia samples are older recorded samples and do not represent pneumonia images from patients with

<sup>1</sup>Chest X-Ray Images (Pneumonia), <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>

<sup>2</sup><https://github.com/AntonisMakris/COVID19-XRay-Dataset>

suspected coronavirus symptoms, while the clinical conditions are missing. Finally, the normal class represents individuals that are not classified as COVID-19 or pneumonia cases. We do not imply that a “normal” patient based on the CXR image does not have any emerging disease.

### 3.2 Performance Metrics

The performance metrics adopted are: Accuracy, Precision, Recall (Sensitivity), F1-score and Specificity. Then, the macro-average results were computed and used to present the classification performance achieved by the networks. *Accuracy* is a commonly used classification metric and indicates how well a classification algorithm can discriminate the classes in the test set. Accuracy can be defined as the proportion of the predicted correct labels to the total number (predicted and actual) of labels. In this study, accuracy refers to the overall accuracy of the model in distinguishing the three classes (covid, pneumonia, normal). *Precision* is the proportion of predicted correct labels to the total number of actual labels while *Recall* is the proportion of predicted correct labels to the total number of predicted labels. Recall is often referred as sensitivity (also called true positive rate). Furthermore, *F1 – score* refers to the harmonic mean of Precision and Recall while *Specificity* (also called true negative rate) measures the proportion of actual negatives that are correctly identified as such.

### 3.3 Transfer learning with CNNs: fine-tuning

Deep learning models require a large amount of data in order to perform accurate feature extraction and classification. Regarding medical data analysis, especially if the disease is at an early stage such as in COVID-19, one major drawback is that the data analyzed were relatively limited. In order to overcome this limitation, transfer learning was adopted. Transfer learning method achieves data training with fewer samples as the retention of the knowledge extracted by a pre-trained model is then transferred to the model to be trained. A pre-trained model is a network that was previously trained on a large dataset, typically on a large-scale image-classification task. The intuition behind transfer learning for image classification is that if a model is trained on a general large dataset, this model will effectively serve in turn as a generic model. The learned features can be used to solve a different but related task involving new data, which usually are of a smaller population to train a CNN from scratch [16]. Thus the need of training from scratch a large model on a large dataset is eliminated.

In this work, the CNN models were fine-tuned to identify and classify the different classes (covid, pneumonia, normal). In fine-tuning, the weights of the top layers of the pre-trained model are “fine-tuned” along with the newly-added classifier layers. Thus, the weights are tuned from generic feature maps to features associated specifically with the provided dataset. The weights used by all CNNs are pre-trained on the ImageNet dataset [5]. Figure 1 illustrates an example of the fine-tuning process on the VGG16 network architecture. The network is instantiated with weights pre-trained on ImageNet. On the top of the figure the layers of the VGG16 network are showed. As stated in 2.1, VGG16 contains 13 convolutional (CONV) and 3 fully-connected (FC) layers. The final set of layers which contain the FC layers along with the *softmax*

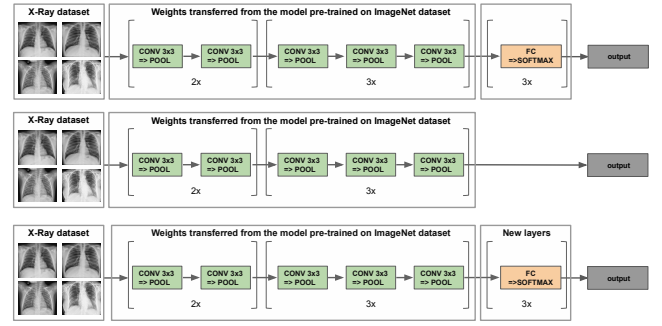


Figure 1: Fine-tuning on the VGG16 network architecture

activation function is called “head”. Afterwards, the FC layers are excluded and the final POOL layer is treated as a feature extractor as depicted in the middle of the figure. Finally, a new FC head layer is randomly initialized and placed on top of the original architecture (bottom of the figure). It is worth mentioning that the body of the network i.e. the CONV layers, have been “repressed” such that only the FC head layer is trained. The reason for this behaviour is that the CONV layers have already learned discriminative filters while FC head layer is randomly initialized from scratch and random values are able to destroy the learned features.

## 4 EXPERIMENTAL EVALUATION

In this research work the effectiveness of several state-of-the-art pre-trained convolutional neural networks was evaluated regarding the detection of COVID-19 disease from chest X-Ray images. More specific, a pool of existing deep learning classifiers were employed namely, VGG16, VGG19, MobileNet V2, Inception V3, Xception, InceptionResNet V2, DenseNet201, ResNet152 V2 and NASNetLarge. The selected models present the highest values in terms of top-5 accuracy which refers to the model’s performance on the ImageNet validation dataset<sup>3</sup>.

### 4.1 Experimental Setup

To train the proposed deep transfer learning models, the python programming language was used including the Keras package and a TensorFlow<sup>4</sup> backend. Keras<sup>5</sup> is a simple to use neural network library built on top of Theano or TensorFlow. This framework was used along with the set of weights learned on ImageNet.

The underlying computing infrastructure that has been used for the execution of the CNNs has been a commodity machine with the following configuration: Ubuntu 18.04 LTS 64-bit; Intel Core i7-8550U CPU @ 1.80GHz × 8; and 16 GiB RAM.

### 4.2 Parameters tuning

All the examined CNNs share some common hyper-parameters. Specifically, all images were scaled to a fixed size of  $224 \times 224$  pixels. The dataset used was randomly split into 80% and 20% for training and testing respectively and the training was conducted for 35 epochs to avoid overfitting for all pre-trained models with a

<sup>3</sup><https://keras.io/api/applications/>

<sup>4</sup>TensorFlow, <https://www.tensorflow.org/>

<sup>5</sup>Keras, <https://keras.io/>

**Table 1: Classification performance obtained from different pre-trained CNN models**

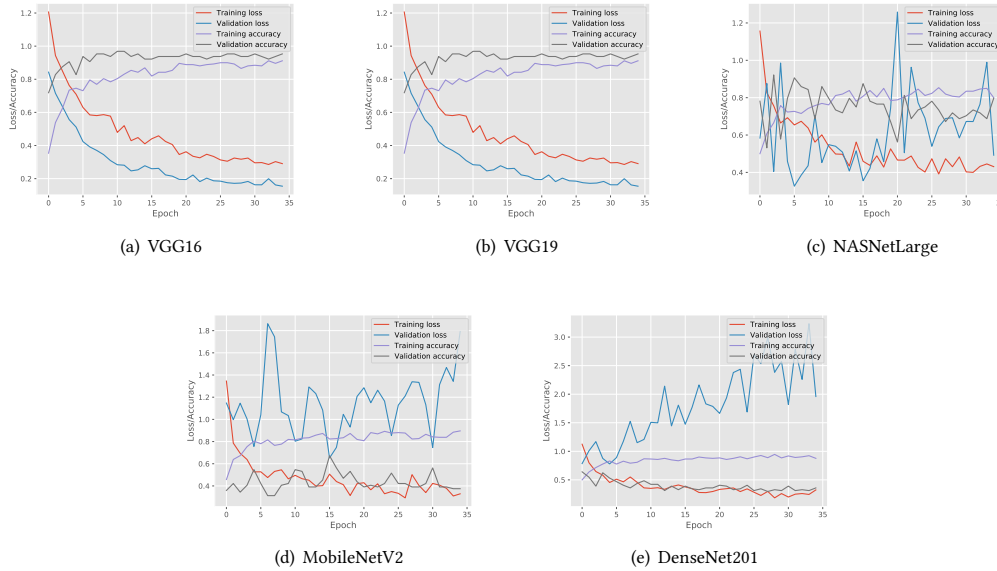
Model	Labels	Precision	Recall	F1-score	Specificity	Overall Accuracy
VGG16	covid	0.96	0.96	0.96	0.98	<b>0.9588</b>
	normal	0.95	1.00	0.98	0.98	
	pneumonia	0.95	0.91	0.93	0.98	
VGG19	covid	1.00	0.92	0.96	1.00	<b>0.9503</b>
	normal	0.95	1.00	0.98	0.98	
	pneumonia	0.95	0.91	0.93	0.95	
MobileNetV2	covid	1.00	0.12	0.21	1.00	<b>0.3970</b>
	normal	1.00	0.10	0.18	1.00	
	pneumonia	0.35	1.00	0.52	0.10	
InceptionV3	covid	1.00	0.04	0.07	1.00	<b>0.5441</b>
	normal	0.74	0.70	0.72	0.89	
	pneumonia	0.46	1.00	0.63	0.43	
Xception	covid	1.00	0.62	0.76	1.00	<b>0.7647</b>
	normal	0.88	0.70	0.78	0.95	
	pneumonia	0.61	1.00	0.76	0.69	
InceptionResNetV2	covid	1.00	0.19	0.32	1.00	<b>0.5588</b>
	normal	0.51	1.00	0.68	0.60	
	pneumonia	0.54	0.59	0.57	0.76	
DenseNet201	covid	1.00	0.12	0.21	1.00	<b>0.3823</b>
	normal	1.00	0.05	0.10	1.00	
	pneumonia	0.34	1.00	0.51	0.08	
ResNet152V2	covid	1.00	0.12	0.21	1.00	<b>0.6176</b>
	normal	0.95	0.90	0.92	0.97	
	pneumonia	0.46	0.95	0.62	0.45	
NASNetLarge	covid	1.00	0.58	0.73	1.00	<b>0.8123</b>
	normal	0.76	0.95	0.84	0.87	
	pneumonia	0.75	0.95	0.84	0.85	

learning rate of  $1e-3$  and a batch size of 8. CNNs were compiled utilizing the optimization method called Adam [11] and all the convolutional layers are activated by the Rectified Linear Unit (ReLU) [13]. Furthermore, a Dropout layer [9] of 0.5 is applied, which means that 50% of neurons will randomly set to zero during each training epoch thus avoiding overfitting on the training dataset. Due to the lack of a larger number of available samples, data augmentation with multiple pre-processing techniques was performed, leveraging Keras *ImageDataGenerator* during fine-tuning. The transformations that employed include random rotation of the images (maximum rotation angle was 30 degrees), horizontal flips, shearing, zooming, cropping and small random noise perturbation. Since the problem comprises 3 classes the “categorical\_crossentropy” is employed as

loss function which compares the distribution of the predictions with the true distribution.

### 4.3 Results & Discussion

In this section the classification performance for each CNN is presented. In order to evaluate the results, the following metrics were adopted for each class (covid, pneumonia, normal): precision, recall (sensitivity), F1-Score, specificity and the overall accuracy of the model as illustrated in Table 1. The results suggest that the VGG16 and the VGG19 achieve the best classification accuracy of 95%. NASNetLarge model showed a moderate accuracy of 81%. The other models did not surpass 80% of accuracy with MobileNetV2 and DenseNet201 presenting the lowest results with 40% and 38% accuracy respectively. Furthermore, the confusion matrices of the



**Figure 2: Accuracy and loss (train and test) of deep learning models**

best two models (VGG16, VGG19), the moderate model (NASNetLarge) and the worst models (MobileNetV2 and DenseNet201) are presented in Figure 3.

A sensitivity of 96% and 92% for the covid class can be observed for VGG16 and VGG19 models respectively. In other words, confirmed positive COVID-19 patients would be accurately identified as “COVID-19 positive” 96% and 92% of the time by employing VGG16 and VGG19 models respectively. Furthermore, the aforementioned models show a high precision value of 96% and 100% for covid class respectively. This implies that for VGG19 there were no classes incorrectly classified as covid from another classes while for VGG16 only one covid case was incorrectly classified as pneumonia as showed in Figures 3(b), 3(a). Another important aspect of the results is the high values associated with specificity. Specifically, the specificity for the covid class is 98% and 100% for VGG16 and VGG19 respectively. This practically means that confirmed negative patients to COVID-19 would be accurately identified as “COVID-19 negative” 98% and 100% of the time using VGG16 and VGG19 models respectively. A similar trend can be depicted in terms of F1-score. Also, one of the very encouraging results is the ability of these models to achieve high sensitivity and precision on the normal class. This ensures that the FPs are minimized not only for the covid but also for the pneumonia class.

Regarding NASNetLarge, a precision of 100% for the covid class is observed which means that there were no normal or pneumonia classes falsely misclassified as covid. Furthermore the model would accurately identify “COVID-19 negative” cases 100% of the time but presents a low sensitivity value. Confirmed COVID-19 cases would be able to be identified almost only half the time. Additionally, the model presents a moderate value of 73% for F1-score in covid class. Indeed, this low value is justified by a large number of FNs. Figure 3(c) depicts that the covid class has 11 misclassified cases in total

as normal or pneumonia. As expected, this is not acceptable when dealing with such a contagious virus.

Although MobileNetV2 and DenseNet201 present the worst results in terms of accuracy, they outperform VGG16 in terms of specificity and precision (100% for both metrics) for covid class. However, as the results suggest, the sensitivity is one of the most important metric in the particular disease. The extremely low value of 12% depicted for both models can have devastating effects regarding virus spread. Only 12% of confirmed COVID-19 cases would accurately identified correctly. Furthermore, the low value of 21% concerning F1-score in covid class implies many FNs. Indeed, both models presents 23 FNs as illustrated in Figures 3(d), 3(e).

Furthermore, we visualized the loss and the accuracy of the same CNNs during their training in Figure 2. Specifically, Figures 2(a), 2(b), 2(c), 2(d) and 2(e) demonstrate the training/validation loss/accuracy of VGG16, VGG19, NASNetLarge, MobileNetV2 and DenseNet201, respectively. The two best models (Figure 2(a) and 2(b)), demonstrate a smooth training process during which the loss gradually decreases and the accuracy increases. Moreover, it can be observed that the accuracy of both training and validation do not deviate much from one another in most cases, a phenomenon that can also be observed for the training and validation loss, indicating that the models do not overfit. On the other hand, the rest of the models not only present a low accuracy, but their validation loss is either increasing or fluctuating. In the case of NASNetLarge (Figure 2(c)), which presents a relatively high accuracy (in the range of 75%), the fluctuating validation loss means that the model most probably overfits. Another interesting fact is that the models with the least number of layers (VGG16 and VGG19) achieve a better classification performance. This can be explained by the fact that neural networks with more hidden layers require more training

data, thus an even larger number of X-Ray samples needs to be provided in these networks.

As illustrated in Table 1 all the CNN models present quite high precision and specificity values in covid class even if the overall accuracy is low, except for VGG16 and VGG19. Confusion matrices confirm this, as the false positives are zero (only 1 FP in VGG16) as illustrated in Figure 3. However, the sensitivity is extremely low in some models. For example, InceptionV3 and DenseNet201 present the lowest values of sensitivity with 4% and 12% respectively. Furthermore, one important observation is that sensitivity presents high values for pneumonia class except in InceptionResNetV2 model. This ensures that patients with common bacterial pneumonia will not missclassified as covid.

Regarding execution times, the most accurate models present the highest values (in seconds). Specifically, NasNetLarge exhibits the highest execution time (4034) followed by VGG19 and VGG16 with 2827 and 2030 respectively. This can be explained by the fact that these models consist of the largest number of parameters. MobileNetV2 presents the lowest execution time (308) and is included along with DenseNet201 (2053) in the models with the worst overall accuracy. Nevertheless, InceptionV3, Xception and InceptionResNetV2 present smaller execution times with 731, 1393 and 1577 respectively, even if the accuracy is much better than DenseNet201.

## 5 CONCLUSION AND FUTURE WORK

In this work, a preliminary study was conducted and presented for the detection of patients positive to COVID-19, a pandemic that infected a large amount of the human population in the first semester of the year 2020. Specifically, the study presented and employed 9 well-known Convolutional Neural Networks (CNNs) for the classification of X-Ray images originating from patients with COVID-19, pneumonia and healthy individuals. Research findings indicated that CNNs have the potential to detect respiratory diseases with high accuracy, although a large amount of sample images needs to be collected. Specifically, VGG16 and VGG19 achieve an overall accuracy of 95%. The high values associated with sensitivity, specificity and precision of covid class, imply the ability of these models to detect positive and/or negative COVID-19 cases accurately. As the results show, determining the most effective model for this classification task involves several performance metrics. Furthermore, one of the very encouraging results is the ability of the aforementioned CNNs to achieve high sensitivity and precision on the normal class thus ensuring the minimization of false positives regarding infection classes. Finally, we would like to emphasize that these methods should not be used directly without clinical diagnosis. For future work, we intend to train the CNNs on more data and to evaluate more architectures for the case of COVID-19 detection.

## REFERENCES

- [1] Asmaa Abbas, Mohammed M Abdelsamea, and Mohamed Medhat Gaber. 2020. Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network. *arXiv preprint arXiv:2003.13815* (2020).
- [2] Ioannis D Apostolopoulos and Tzani A Mpesiana. 2020. Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. *Physical and Engineering Sciences in Medicine* (2020), 1.
- [3] François Chollet. 2017. Xception: Deep Learning with Depthwise Separable Convolutions. In *2017 IEEE Conference on Computer Vision and Pattern Recognition*,

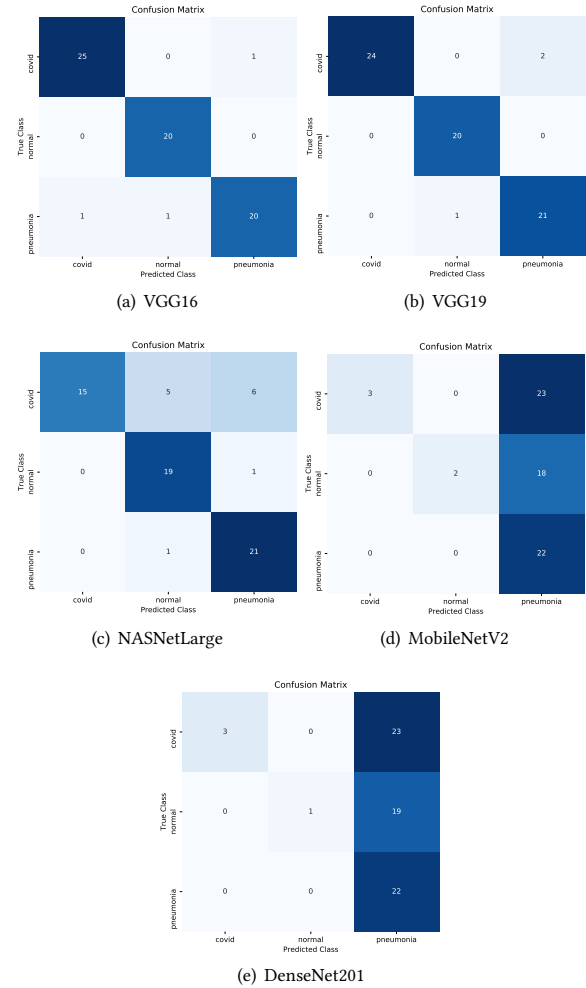


Figure 3: Confusion matrix of deep learning models

CVPR 2017, Honolulu, HI, USA, July 21–26, 2017. IEEE Computer Society, 1800–1807. <https://doi.org/10.1109/CVPR.2017.195>

- [4] Joseph Paul Cohen, Paul Morrison, and Lan Dao. 2020. COVID-19 image data collection. *arXiv 2003.11597* (2020). <https://github.com/ieee8023/covid-chestxray-dataset>
- [5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.
- [6] Muhammad Farooq and Abdul Hafeez. 2020. Covid-resnet: A deep learning framework for screening of covid19 from radiographs. *arXiv preprint arXiv:2003.14395* (2020).
- [7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016*. IEEE Computer Society, 770–778. <https://doi.org/10.1109/CVPR.2016.90>
- [8] Ezz El-Din Hemdan, Marwa A Shouman, and Mohamed Esmail Karar. 2020. Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images. *arXiv preprint arXiv:2003.11055* (2020).
- [9] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580* (2012).
- [10] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. 2017. Densely Connected Convolutional Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2261–2269.

- [11] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
- [12] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.
- [13] Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*. 807–814.
- [14] Ali Narin, Ceren Kaya, and Ziyne Pamuk. 2020. Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. *arXiv preprint arXiv:2003.10849* (2020).
- [15] L. D. Nguyen, D. Lin, Z. Lin, and J. Cao. 2018. Deep CNNs for microscopic image classification by exploiting transfer learning and feature concatenation. In *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*. 1–5.
- [16] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22, 10 (2009), 1345–1359.
- [17] Waseem Rawat and Zenghui Wang. 2017. Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. *Neural Computation* 29 (06 2017), 1–98. [https://doi.org/10.1162/NECO\\_a\\_00990](https://doi.org/10.1162/NECO_a_00990)
- [18] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen. 2018. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4510–4520.
- [19] Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR* abs/1409.1556 (2015).
- [20] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. 2017. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence* (San Francisco, California, USA) (AAAI’17). AAAI Press, 4278–4284.
- [21] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. 2016. Rethinking the Inception Architecture for Computer Vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016*. IEEE Computer Society, 2818–2826. <https://doi.org/10.1109/CVPR.2016.308>
- [22] C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. 2015. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1–9.
- [23] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc Le. 2018. Learning Transferable Architectures for Scalable Image Recognition. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 8697–8710. <https://doi.org/10.1109/CVPR.2018.00907>