A Comparison of Deep Learning Models for Detecting COVID-19 in Chest X-ray Images

Enrique Peláez [*] **Ricardo Serrano** [*] **Geancarlo Murillo** [*]
Washington Cárdenas [**]

[*] *Escuela Superior Politécnica del Litoral - ESPOL University, Electrical and Computer Engineering, Guayaquil, Ecuador, (e-mail: epelaez@espol.edu.ec, ricserra@espol.edu.ec, geanmuri@espol.edu.ec)*
[**] *Escuela Superior Politécnica del Litoral - ESPOL University, Life Science, Guayaquil, Ecuador, (e-mail: wcardenas@espol.edu.ec)*

Abstract: COVID-19 has spread around the world rapidly causing a pandemic. In this research, a set of Deep Learning architectures, for diagnosing the presence or not of the disease have been designed and compared; such as, a CNN with 4 incremental convolutional blocks; a VGG-19 architecture; an Inception network; and, a compact CNN model known as MobileNet. For the analysis and comparison, transfer learning techniques were used in forty-five different experiments. All four models were designed to perform binary classification, reaching an accuracy above 95%. A set of different scores were implemented to compare the performance of all models, showing that the VGG-19 and Inception configurations performed the best.

Keywords: Machine Learning, Transfer Learning, Convolutional Neural Network, X-Ray Image Analysis, COVID-19.

1. INTRODUCTION

During the last year, a family of the Coronavirus has been the cause of millions of fatalities around the world [1]. As reported by Punn et al. (2020), the first known epidemic, caused by this family of viruses, were the Severe Acute Respiratory Syndrome (SARS) and the Middle East Respiratory Syndrome (MERS), which appeared in 2003 and 2012 respectively. In December 2019 a new Coronavirus, SARS-CoV-2, appeared in Wuhan [2], which spreads rapidly, as reported by WHO[2], causing a pandemic, as it was declared in March 11, 2020. Since its appearance in December 2019, up until this paper is written, 4th of May 2021, this virus has infected more than 150 million inhabitants around the globe and has caused the death of more than 3.2 million people worldwide, as reported by WHO (2021). By this date in countries like Ecuador, the Ministry of Public Health -Ministerio (2021), also reported more than 387 thousand confirmed COVID-19 cases, and more than 18.7 thousand of them had died.

Given this overwhelming situation, a rapid diagnosis of the disease would have been beneficial for timely treatment, as well as for following the appropriate quarantine protocols; and so, reducing the spread of the virus. Today, these diagnoses are made in a relatively short time, however the costs associated with acquiring the testing kits, or the chemical reagents, or accessing specialized equipment including the technicians, still is a financial challenge, for both governments and citizens, particularly in countries with limited public health systems, Apostolopoulos and Mpesiana (2020).

Ai et al. (2020) showed that COVID-19 pneumonia looks different, from a radiologist's perspective; hence, chest X-ray and computed tomography (CT) imaging are becoming useful tools for screening this disease, due to their high sensitivity to structures containing air and fluid, as noted by Hussain et al. (2020), which make them a useful choice for researchers, to attempt to automate the diagnosing process of COVID-19.

Machine learning (ML) techniques have shown effective results in recognizing patterns in clinical images, and Convolutional Neural Networks (CNN) in particular, have outperformed conventional ML techniques, effective in recognizing complex objects and patterns, segmenting lesions and extracting features from them, as proposed in Greenspan et al. (2016), with a growing trend in clinical image analysis and artificial vision applications in health services.

Deep Learning (DL) techniques have been used for detecting COVID-19 in X-ray chest images; as reported by Ghaderzadeh and Asadi (2021) CNN architectures, such as the VGG, ResNet, MobileNet, GoogleNet and Inception, have shown promising results and have been considered as the base models for building rapid diagnostic decision-making tools and increase the diagnostic accuracy.

In this research, four DL models for diagnosing COVID-19 were analyzed: a CNN model with incremental blocks, and 3 pre-gated architectures (VGG-19, MobileNetV2 and InceptionV3), which were trained to analyze chest radiog-

[1] https://coronavirus.jhu.edu/data/mortality
[2] https://www.who.int/news-room/detail/27-04-2020-who-timeline–covid-19

raphy images to produce a diagnosis with a probability. This paper is organized as follows: section 2 describes current developments of DL models, designed for recognizing the patterns related to COVID-19; section 3 shows the methodology followed; section 4 presents the models and their architectures; section 5 shows the results and a discussion; and, section 6 draws the main conclusions.

2. THE CONTEXT, METHODOLOGY AND THE DATASET

CNNs are end-to-end DL models for artificial vision, with high level performance, in particular in medical image analysis. COVID-19 still is a challenge to health service providers and to DL techniques, therefore X-ray images have become suitable for training deep learning models. In the early days of the outbreak, CT scans were common, but over time, the X-rays images have also become available and easier to access in public repositories, as reported in Ghaderzadeh and Asadi (2021).

Recently, there have been several proposals for detecting COVID-19 on chest X-ray images, both as binary or multiclass classification, the GoogLeNet or Inception model developed by Szegedy et al. (2014); ResNet by He et al. (2015); VGG by Simonyan and Zisserman (2014); MobileNet by Pan et al. (2020) and others.

Other approaches look to speed up the COVID-19 diagnosis; such as, the feature extraction model proposed by Tang et al. (2020), based on Random Forest (RF) to classify the severity of the virus; or a model based on the analysis of cough using smart phones, presented by Imran et al. (2020); or a CNN model with a U-net architecture for segmenting lung lesions, which allows capturing better contextual relationships, as discussed in Zhou et al. (2020).

Aggarwal et al. (2021) and Jain et al. (2021) have also compared DL architectures using images, categorized as normal, COVID and Pneumonia, with transfer learning and data augmentation for increasing the number of samples. In this research we tested 4 DL models, 3 of them were trained using transfer learning, and one trained from scratch to compare the performance of the models.

2.1 Methodology

The analysis of our models and the comparison of results were performed in three stages: data collection from relevant sources and selection of reliable X-ray images; pre-processing to comply with the expected input format in DL networks; and the design and definition of hyperparameters of our DL models.

During the first stage, data collection was focused on finding and selecting X-ray chest images from confirmed COVID-19 cases, as well as cases with negative diagnosis. During pre-processing we applied techniques for cleaning, debugging and augmenting the set of images; and, during the design stage, DL models' architectures and hyperparameters were defined, as well as the evaluation metrics, which we used to determine the best model's performance.

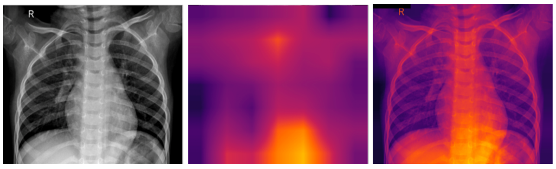

Fig. 1. non-COVID-19 X-ray

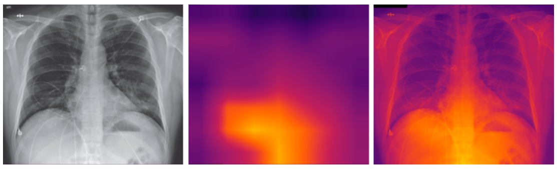

Fig. 2. COVID-19 X-ray

2.2 Data collection and validation of images

In this stage we explored several medical repositories specialized in medical images, in particular for X-ray chest images; among the publicly available repositories we reviewed Kaggle, Mendeley and various Github sites in which researchers collect images from medical organizations, such as the Italian Society of Medical and Interventional Radiology, SIRM (2020); the New England Journal of Medicine, NEJM (2020); and, the Radiological Society of North America, RSNA (2020). Additionally, we have collected Ecuadorian X-ray images of patients diagnosed as positive for COVID-19, from a local hospital in Guayaquil, a city that was the focus of the pandemic in Ecuador.

The data collection stage ended up with 20866 X-ray images, 5414 images correspond to COVID-19 confirmed cases, and 11632 to non-COVID-19 or negative cases (the rest corresponded to other types of lung pathogens, not related to COVID-19). To ensure the images, collected from different sources, correspond to chest X-ray images, they were verified manually, as a result we obtained a balanced dataset comprised of 3206 images of confirmed COVID-19 cases, and 3206 images for non-COVID-19. A sample of these images can be seen in Figures 1 and 2. This dataset is referred as Dataset_1.

2.3 Pre-processing

The collected images came from different sources, with a variety of characteristics, such as different sizes and resolution, variations of light intensity and fuzziness, hence it was necessary to perform pre-processing procedures, such as: resizing to a standard 224x224 size; smoothing and histogram equalization to eliminate noise and highlight the contrast. For smoothing, a median 5x5 kernel was used. To make sure the images were no duplicated, cleaning and debugging was necessary, as well as to check each image to avoid non chest X-ray images. For this procedure we use a technique called Image Hashing, as introduced by Buchner (2020). This process was applied to Dataset_1, to produce the Dataset_2.

The Image Data Generator from Keras was also used for data augmentation to increase the number of images; 12 new images were created per sample. Table 1, shows the combinations of data augmentation techniques and parameters used. This process was applied to Dataset_1, producing the Dataset_3.

Table 1. Data Augmentation Parameters

Technique	1	2	3
Horizontal Flip	True	True	True
Width Shift Range	0.2	0.1	0.1
Height Shift Range	0.2	0.1	0.1
Shear Range	0.2	0.1	Don't Apply
Zoom Range	0.2	0.1	Don't Apply
Rotation Range	20	10	Don't Apply

3. DEEP LEARNING MODELS DESIGN

During this stage, different architectures were designed based on a CNN model, then compared their classification performance and their ability to diagnose the presence, or not of COVID-19; hence, a set of hyper-parameters and metrics were defined for evaluating the results.

For selecting the best CNN architecture, 4 different models were designed and built; one was conceived from scratch based on Convolutional Blocks, and the others were pre-trained based on a transfer-learning approach, speeding up the training process and searching for minimum error.

X-ray images from COVID-19 patients are captured in gray scale and contain several patterns, spread throughout the lungs, as they can be observed in Figure 2. Learning the disease's patterns from the set of training images, requires an architecture with enough processing units to cope with the number of features expected to discriminate the patterns associated with the disease.

Pre-trained models have been applied to extract relevant features and patterns from images, with promising results as discussed by Zhuang et al. (2020), an approach we used to speed up the extraction of relevant features and patterns from X-ray gray images. In this research, the pre-trained models used were: a) VGG-19, b) MobileNet V2, and c) Inception V3; these based models were trained as feature extractors. For the classification phase we designed our own Multi Layer Perceptron (MLP), to classify an X-ray image as COVID or Non-COVID. In this research a set of experiments were designed to test the models and cross-validate their results.

Following the transfer learning approach, the convolutional section of our CNN models were trained to extract the X-ray features. The transferred model's parameters matrices were used as initialization of our parameters, prior to train the models. Our CNN's classification phase were also trained to discriminate from those extracted features. The forth proposed CNN model, with 4 incremental convolutional blocks, was trained from scratch without using transfer learning, showing comparable results in some of the performance metrics, as it is described in section 4.

Regarding the hyper-parameter settings, all models share the same configuration. The convolutional layers are activated by a Rectified Linear Unit (ReLU); a binary-crossentropy loss function; and, a Stochastic Gradient Descent optimizer. The MLP has two hidden layers with a Dropout layer. All models were also compiled using the Adam optimization method. A description of each architecture follows:

VGG-19. This architecture proposed by Simonyan and Zisserman (2014) is composed by 19 trainable layers, which includes convolutional blocks, max-pooling layers, a fully connected layer, one dropout, and one dense connected layer at the output, as it is shown in Figure 3.

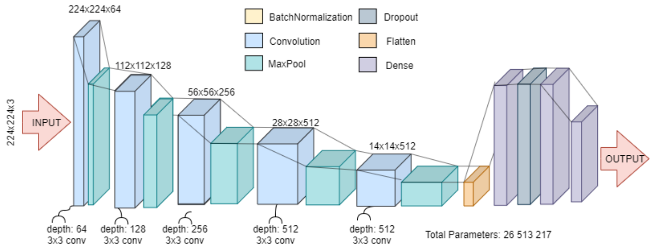

Fig. 3. VGG-19 Architecture

MobileNet V2. Developed by Szegedy et al. (2014), this model belongs to the family of general-purpose artificial vision neural networks, designed for mobile devices. This model uses Depthwise Separable Convolutions and introduces the concepts of Inverted Residuals and linear Down-sampling, resulting in an efficient memory improvement. This model's architecture is comprised of fully convolution layers with 32 filters, 19 residual down-sampling, dropout layers, and batch normalization layers, as shown in Figure 4.

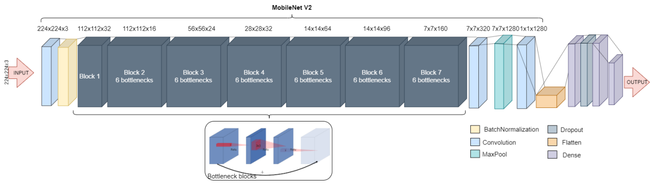

Fig. 4. MobileNet V2 Architecture

Inception V3. This model, first developed by GoogLeNet, represents the implementation of many ideas developed by various researchers over the years, as discussed in Szegedy et al. (2016). The model, as represented in Figure 5, consists of symmetric and asymmetric series of convolutional blocks, avgPool and maxPool poolings, concatenations, dropouts, and fully connected layers with a Softmax activation function at the output.

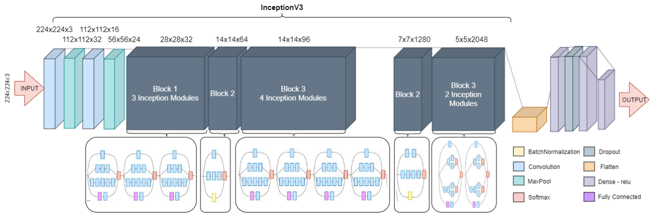

Fig. 5. Inception V3 Architecture

A CNN with 4 incremental Conv blocks. In this model 4 incremental conv blocks are used, each with 8 conv layers, 32 filters of size 3x3, and Batch Normalization after each conv layer to normalize every batch of data. Then a max-pooling layer with 2x2 filters, after every 2 conv layers, which is used to reduce the dimensionality by half. In this architecture there is an incremental 10% dropout layer, after every two convolutional layers, starting from a 20%

dropout. Figure 6 shows the main components of such a network.

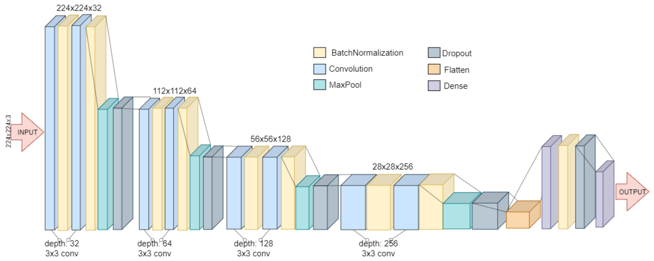

Fig. 6. CNN with 4 incremental Convolutional Blocks Architecture

Experiments. Three sets of experiments were designed for training the models based on the datasets, Dataset_1, plus two variations: Dataset_2 containing the pre-processed images; and, Dataset_3 with data augmentation. All datasets were fragmented in 80% for training, 20% for testing and in each experiment the following hyper-parameters were tested:

Table 2. Experiments

Exp.	Model	Hyper-params	Dataset
1	VGG-19 MobileNet V2 Inception V3	Keras default config ReLu Binary Crossentropy Adam Optimizer	Dataset_1 3206 COVID 3206 non-COVID
	CNN 4ICB	ReLu HeUniform Binary Crossentropy SGD Optimizer Alpha Momentum	
2	VGG-19 MobileNet V2 Inception V3	Same as exper. 1	Dataset_2 3206 COVID 3206 non-COVID
	CNN 4ICB		
3	VGG-19 MobileNet V2 Inception V3	Same as exper. 1	Dataset_3 38472 COVID 38472 non-COVID
	CNN 4ICB		

Performance Metrics. To evaluate the performance of each model, a set of metrics were defined, such as the Confusion Matrix, the ROC Curve and the AUC values, as well as the scores that measure Precision & Recall, the F1 Score, and the set of Support.

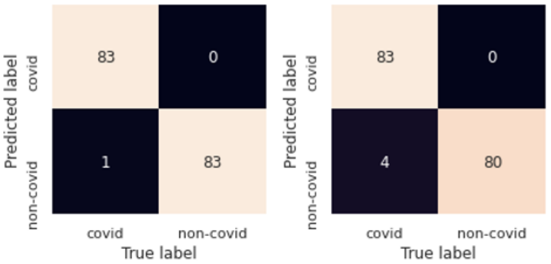

Fig. 7. Confusion Matrix for the VGG-19 and MobileNet V2

4. ANALYSIS OF RESULTS

As it is shown in Table 2, training was performed through 3 sets of different experiments, with 15 different configurations each: In the first experiment all models were trained using Dataset_1, with the 3206 original chest X-ray images of confirmed COVID-19 cases, and 3206 non-COVID-19 cases. The second experiment was performed with Dataset_2; and, the third with Dataset_3, which included data augmentation images, generated from the Dataset_1.

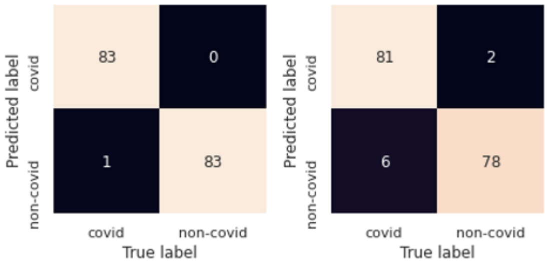

Fig. 8. Confusion Matrix for the Inception V3 and CNN Incremental Convolutional Blocks

In general, Experiments 1 and 2 presented better results with the VGG-19, MobileNet V2 and Inception V3 architectures, while the CNN with 4 incremental Conv blocks performed the worst in discriminating between the two classes. Experiment 3 was contradictory for the VGG-19, MobileNet V2 and Inception models, however the CNN with 4 incremental Conv blocks improved slightly. The pre-processed versions of these datasets did not show significantly better results in any of the experiments, as compared to the training with Dataset_1. The third experiment, which included data augmentation, showed the best results for the three models, as compared to the other configurations. The confusion matrices for the four models are shown in Figure 7 and Figure 8.

After 60 trainings cycles, the ROC curve and confusion matrices showed that VGG-19 and Inception V3 performed best, with 0.994 and 0.988 AUC respectively, as compared to the other 2 models, as seen in Figure 9.

As it can be observed from the confusion matrices, VGG-19 and Inception V3 performed the best, with 83 true negatives and true positives out of 84 instances, and 1 false negative in both cases; as for the other models, the MobileNet V2 and the CNN with 4 incremental convolutional blocks, there were 2 and 3 false negatives respectively; and 4 false positive in the case of the CNN with 4 incremental conv blocks.

Table 3, shows the scores for Precision, Recall and F1 metrics, which corroborate the best performance of the VGG-19 and Inception V3 models, tested on the set of

Table 3. Experiment Results

Model		Precision	Recall	F1-Score	Support
VGG-19	C-19	0.99	1	0.99	83
	noC-19	1	0.99	0.99	84
MobileNet2	C-19	0.95	1	0.98	83
	noC-19	1	0.95	0.98	84
Inception3	C-19	0.99	1	0.99	83
	noC-19	1	0.99	0.99	84
CNN 4BI	C-19	0.93	0.98	0.95	83
	noC-19	0.97	0.93	0.95	84

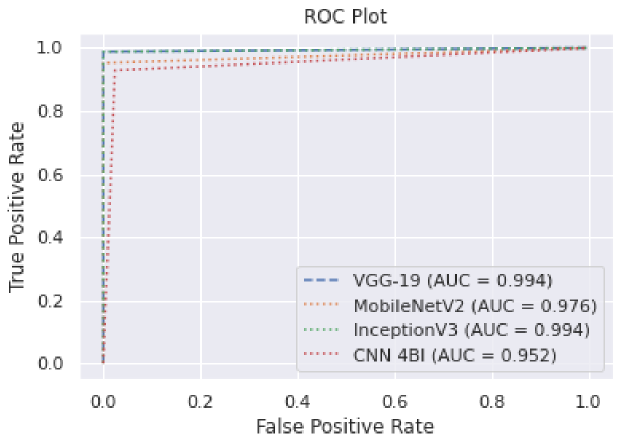

Fig. 9. ROC Curves

84 instances for predicting COVID-19 and non-COVID-19 cases.

Implementation and evaluation of models. For implementing the models we used Google Colab's free GPU, NVIDIA Tesla K80 12 GB. The models, pre-processing and training were implemented in TensorFlow 2.3.0, running in Python 3.6.9. VGG-19, Inception V3 and MobileNet V2 models showed comparable results; however, there were differences in terms of training time and the number of parameters to learn. With the confusion matrix results, accuracy, sensitivity and specificity were evaluated using (1), (2) and (3) respectively. Tables 4 and 5 show a summary of these results:

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \qquad (1)$$

$$Sensitivity = TP/(TP + FN) \qquad (2)$$

$$Specificity = TN/(TN + FP) \qquad (3)$$

Table 4. Performance Metrics and Scores

Model	Accuracy	Sensivity	Specificity	Training Time
VGG-19	0.994	1.000	0.988	8min 52s
MobileNet V2	0.976	1.000	0.952	8min 32s
Inception V3	0.994	1.000	0.988	8min 44s
CNN 4BI	0.952	0.976	0.929	18min 57s

Table 5. Trainable and Non-trainable params

Model	Total Params	Trainable Params	Non-trainable Params
VGG-19	26,513,217	6,488,833	20,024,384
MobileNetV2	18,380,609	16,122,625	2,254,984
InceptionV3	34,976,289	13,173,505	21,802,784
CNN 4IB	7,599,393	7,597,217	2,176

MobileNet V2 was the model with the largest number of parameters to be learned; in contrast, the VGG-19 model had the fewest number of parameters, although the training time was not significantly different among the models. The training time, for the three models where transfer learning was used, was around 8 minutes, as it is shown in Table 4, except for CNN with 4 incremental conv blocks model, which needed twice the time to be trained.

The models with transfer learning were trained during 30 epochs. Figures 10 through 13 show the training and validation recorded at each training cycle; the left images show the training and validation accuracy and the right images show the corresponding loss during training and validation procedures, at every epoch for each model tested.

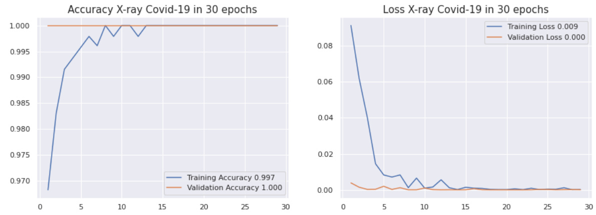

Fig. 10. Training and validation accuracy per epoch on VGG-19

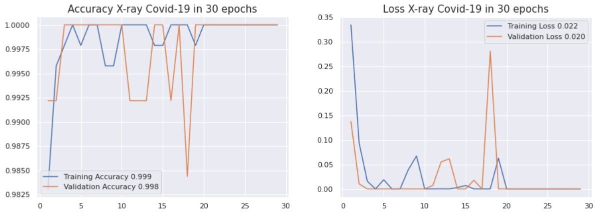

Fig. 11. Training and validation accuracy per epoch on MobileNet V2

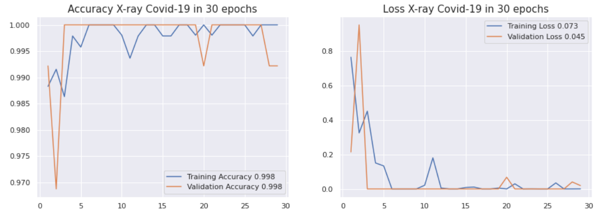

Fig. 12. Training and validation accuracy per epoch on Inception V3

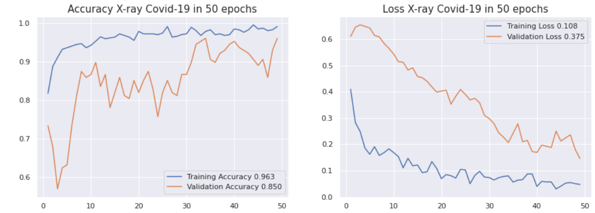

Fig. 13. Training and validation accuracy per epoch on CNN

The VGG-19 architecture compared to the other models, converged after 10 epochs, as Figure 10 shows, a stable condition was reached at an average loss of 0.009. Figure 11, on the other hand, shows that the MobileNet V2 model reached stability after 20 epochs, with an average loss of 0.022. A similar behavior was observed for the Inception V3 model, it took 15 epochs to reach stability, with an average loss of 0.108, as it can be seen in Figure 12.

While the model CNN with 4 incremental convolutional blocks, shown in Figure 13, needed more than 40 epochs to reach stability, with an average loss of 0.025.

5. CONCLUSIONS

This work presents three deep learning models based on pre-trained CNN architectures (VGG-19, MobileNet V2 and Inception V3) and one fully customized CNN with 4 incremental convolutional blocks, trained to learn patterns of COVID-19 pneumonia in chest X-ray images. These models were trained with three different datasets applying various techniques of image pre-processing and data augmentation in order to obtain improvements in the learning process showing promising accuracy and stability in loss.

All four models were designed to perform binary classification for COVID-19 and non-COVID-19 cases (other types of lung diseases will be classified as non-COVID-19), and they reached an accuracy above 95%, as in the with 4 incremental convolutional blocks model, but close to 100% in the VGG-19 and Inception V3 models, with high sensitivity and specificity.

The accuracy of the models proposed could be improved with a greater number of chest X-ray images of COVID-19 and non-COVID-19 cases. We observed that in this case, applying data augmentation to chest X-ray images does not necessarily improve the models' results.

This research has proposed an alternative methodology and tools for diagnosing this new disease, using conventional chest X-ray images, as they are available in countries with limited infrastructure for medical image analysis, through deep learning models based on convolutional neural networks, which can be provided as a remote service through a web system and support the medical community.

REFERENCES

Aggarwal, S., Gupta, S., Alhudhaif, A., Koundal, D., Gupta, R., and Polat, K. (2021). Automated covid-19 detection in chest x-ray images using fine-tuned deep learning architectures. *Expert Systems*, e12749.

Ai, T., Yang, Z., Hou, H., Zhan, C., Chen, C., Lv, W., Tao, Q., Sun, Z., and Xia, L. (2020). Correlation of chest ct and rt-pcr testing for coronavirus disease 2019 (covid-19) in china: A report of 1014 cases. *Radiology*, 296, 200642. doi:10.1148/radiol.2020200642.

Apostolopoulos, I.D. and Mpesiana, T.A. (2020). Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. *Physical and Engineering Sciences in Medicine*, 43(2), 635–640. doi:10.1007/s13246-020-00865-4.

Buchner, J. (2020). Imagehash. URL https://github.com/JohannesBuchner/imagehash.

Ghaderzadeh, M. and Asadi, F. (2021). Deep learning in the detection and diagnosis of covid-19 using radiology modalities: a systematic review. *Journal of Healthcare Engineering*, 2021.

Greenspan, H., van Ginneken, B., and Summers, R.M. (2016). Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique. *IEEE Transactions on Medical Imaging*, 35(5), 1153–1159. doi:10.1109/TMI.2016.2553401.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep Residual Learning for Image Recognition. *arXiv e-prints*, arXiv:1512.03385.

Hussain, S., Sohail, A., Zafar, M., and Khan, A. (2020). Coronavirus disease analysis using chest x-ray images and a novel deep convolutional neural network.

Imran, A., Posokhova, I., Qureshi, H.N., Masood, U., Riaz, M.S., Ali, K., John, C.N., Hussain, M.I., and Nabeel, M. (2020). Ai4covid-19: Ai enabled preliminary diagnosis for covid-19 from cough samples via an app. *Informatics in Medicine Unlocked*, 20, 100378. doi:10.1016/j.imu.2020.100378.

Jain, R., Gupta, M., Taneja, S., and Hemanth, D.J. (2021). Deep learning based detection and analysis of covid-19 on chest x-ray images. *Applied Intelligence*, 51(3), 1690–1700.

Ministerio, M.d.S.P. (2021). Actualización de casos de coronavirus en ecuador. URL https://www.salud.gob.ec/coronavirus-covid-19/.

NEJM (2020). The New England Journal of Medicine.

Pan, H., Pang, Z., Wang, Y., Wang, Y., and Chen, L. (2020). A new image recognition and classification method combining transfer learning algorithm and mobilenet model for welding defects. *IEEE Access*, 8, 119951–119960. doi:10.1109/ACCESS.2020.3005450.

Punn, N.S., Sonbhadra, S.K., and Agarwal, S. (2020). Covid-19 epidemic analysis using machine learning and deep learning algorithms. doi:10.1101/2020.04.08.20057679.

RSNA (2020). Radiological Society of North America.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556.

SIRM (2020). Società Italiana di Radiologia Medica e Interventistica. URL https://sirm.org.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2014). Going Deeper with Convolutions. *arXiv e-prints*.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem, 2818–2826. doi:10.1109/CVPR.2016.308.

Tang, Z., Zhao, W., Xie, X., Zhong, Z., Shi, F., Liu, J., and Shen, D. (2020). Severity assessment of coronavirus disease 2019 (covid-19) using quantitative features from chest ct images.

WHO, W.H.O. (2021). Weekly epidemiological update - 5 january 2021.

Zhou, T., Canu, S., and Ruan, S. (2020). An automatic covid-19 ct segmentation based on u-net with attention mechanism.

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., and He, Q. (2020). A comprehensive survey on transfer learning.