

**UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ**

**Colegio de Ciencias e Ingenierías**

**COVID-19 Resnet: Residual Neural Network for COVID-19  
Classification with Bayesian Data Augmentation**

**Javier Sebastián Balseca Zurita  
Martín Alejandro Cruz Patiño**

**Ingeniería Industrial**

Trabajo de fin de carrera presentado como requisito  
para la obtención del título de  
INGENIERO INDUSTRIAL

Quito, 17 de mayo de 2021

**Universidad San Francisco de Quito USFQ**  
**Colegio de Ciencia e Ingenierías**

**HOJA DE CALIFICACIÓN**  
**DE TRABAJO DE FIN DE CARRERA**

**COVID-19 Resnet: Residual Neural Network for COVID-19 Classification  
with Bayesian Data Augmentation**

**Javier Sebastián Zurita Balseca**  
**Martin Alejandro Cruz Patiño**

**Nombre del profesor, Título académico**

**María Gabriela Baldeón Calisto, PhD**

Quito, 17 de mayo de 2021

## © DERECHOS DE AUTOR

Por medio del presente documento certifico que he leído todas las Políticas y Manuales de la Universidad San Francisco de Quito USFQ, incluyendo la Política de Propiedad Intelectual USFQ, y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo quedan sujetos a lo dispuesto en esas Políticas.

Asimismo, autorizo a la USFQ para que realice la digitalización y publicación de este trabajo en el repositorio virtual, de conformidad a lo dispuesto en la Ley Orgánica de Educación Superior del Ecuador.

Nombres y apellidos: Javier Sebastián Balseca Zurita

Código: 00132639

Cédula de identidad: 1717999674

Nombres y apellidos: Martín Alejandro Cruz Patiño

Código: 00126205

Cédula de identidad: 1724022734

Lugar y fecha: Quito, 17 de mayo de 2021

## **ACLARACIÓN PARA PUBLICACIÓN**

**Nota:** El presente trabajo, en su totalidad o cualquiera de sus partes, no debe ser considerado como una publicación, incluso a pesar de estar disponible sin restricciones a través de un repositorio institucional. Esta declaración se alinea con las prácticas y recomendaciones presentadas por el Committee on Publication Ethics COPE descritas por Barbour et al. (2017) Discussion document on best practice for issues around theses publishing, disponible en <http://bit.ly/COPETHeses>.

## **UNPUBLISHED DOCUMENT**

**Note:** The following capstone project is available through Universidad San Francisco de Quito USFQ institutional repository. Nonetheless, this project – in whole or in part – should not be considered a publication. This statement follows the recommendations presented by the Committee on Publication Ethics COPE described by Barbour et al. (2017) Discussion document on best practice for issues around theses publishing available on <http://bit.ly/COPETHeses>.

## RESUMEN

La COVID-19 es una enfermedad infecciosa causada por un nuevo coronavirus llamado SARS-CoV-2. El primer caso apareció en diciembre del 2019 y hasta el momento sigue representando un gran desafío a nivel mundial. La precisa detección del virus en pacientes COVID-19 positivos es un paso crucial para reducir la propagación de esta enfermedad altamente contagiosa. En este trabajo se implementa una red neuronal residual convolucional (ResNet) para el diagnóstico automatizado de la COVID-19. La ResNet implementada puede clasificar la radiografía del tórax de un paciente en COVID-19 positivo, neumonía causada por otro virus o bacteria, y paciente saludable. Además, para aumentar la precisión del modelo y superar la escasez de imágenes médicas en el set de entrenamiento, se aplica una estrategia de aumento de datos personalizada utilizando la optimización bayesiana en tres pasos. La ResNet propuesta alcanza un 94% de precisión, 95% de sensibilidad y 95% en el F1-score en set de prueba. Adicionalmente, presentamos las operaciones de aumento de datos que ayudaron a incrementar el rendimiento de la red neuronal y que pueden ser utilizados por otros investigadores en el desarrollo de modelos para la clasificación de imágenes médicas.

**Palabras Clave:** Red neuronales convolucionales, Optimización bayesiana, Análisis de imágenes médicas, hiperparámetros.

## ABSTRACT

COVID-19 is an infectious disease caused by a novel coronavirus called SARS-CoV-2. The first case appeared in December 2019, and until now it still represents a significant challenge to many countries in the world. Accurately detecting positive COVID-19 patients is a crucial step to reduce the spread of the disease, which is characterized by a strong transmission capacity. In this work we implement a Residual Convolutional Neural Network (ResNet) for an automated COVID-19 diagnosis. The implemented ResNet can classify a patient's Chest-Xray image into COVID-19 positive, pneumonia caused from another virus or bacteria, and healthy. Moreover, to increase the accuracy of the model and overcome the data scarcity of COVID-19 images, a personalized data augmentation strategy using a three-step Bayesian hyperparameter optimization approach is applied to enrich the dataset during the training process. The proposed COVID-19 ResNet achieves a 94% accuracy, 95% recall, and 95% F1-score in test set. Furthermore, we also provide an insight into which data augmentation operations are successful in increasing a CNNs performance when doing medical image classification with COVID-19 CXR.

**Keywords:** Convolutional neural networks, Deep learning, medical image analysis, Data augmentation, hyperparameters.

## TABLE OF CONTENTS

<b>INTRODUCTION.....</b>	<b>10</b>
<b>TOPIC DEVELOPMENT.....</b>	<b>15</b>
Literature Review.....	15
Methodology.....	16
Residual neural network.....	16
<b>Data augmentation.....</b>	<b>20</b>
Bayesian hyperparameter optimization for data augmentation.....	22
Experimental Setup.....	25
Dataset.....	25
Preprocessing operations.....	26
Gray scale conversion.....	27
Reseizing.....	27
Adaptative histogram equalization.....	27
Training.....	27
<b>RESULTS .....</b>	<b>28</b>
Bayesian optimization results.....	28
Classification results.....	29
Benchmark Comparison.....	31
<b>CONCLUSIONS .....</b>	<b>32</b>
<b>REFERENCES.....</b>	<b>34</b>

**TABLES INDEX**

Table 1. Hyperparameters and search bounds optimized in each of the three steps.....	24
Table 2. Number of CXR images per category for training, validation, and testing. ....	26
Table 3. Best hyperparameter values found.....	29
Table 4. Hyperparameters tested based on literature. ....	29
Table 5. Classification results for each category in the test set. ....	31
Table 6. Evaluation metrics of the implemented COVID-19 ResNet (*).....	31



**FIGURES INDEX**

Figure 1. Examples of CXR images. ....	12
Figure 2. Basic residual block for the COVID-19 ResNet architecture. ....	18
Figure 3. The COVID-19 ResNet architecture.. ....	20
Figure 4. Data augmentation operations applied to a CXR image in the training dataset. ....	22
Figure 5. Examples of a preprocessed CXR .....	26

## INTRODUCTION

COVID-19 is a disease caused by a new coronavirus called SARS-Cov-2. The first reports of this new virus came from Wuhan, Republic of China. The most common symptoms associated with COVID-19 are fever, fatigue, and dry cough. Meanwhile, the most severe symptoms include shortness of breath, confusion, pressure in the chest and loss of appetite [1]. In comparison with the severe acute respiratory syndrome (SARS) and the Middle East respiratory syndrome (MERS), the SARS-Cov-2 is characterized by a lower mortality rate but a stronger transmission capacity [2].

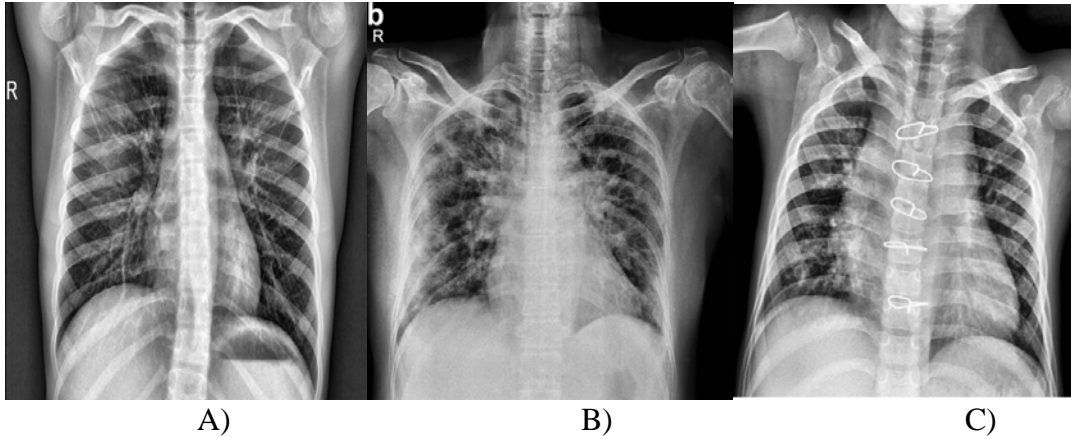
The initial screening methods for COVID-19 diagnosis are the Real-Time reverse transcription-Polymerase Chain Reaction test (RT-PCR), antibody test (serology), and auxiliary diagnosis tests like computed tomography (CT) and chest X-ray (CXR) [2]. The RT-PCR test is recommended for people who have shown symptoms of the Coronavirus, people who have been in contact with a confirmed case, and people who have been traveling or participating in social events [3]. On the other hand, the antibody test is suggested to people who believe have had the virus in the past as it looks for the antibodies in the blood [4].

The laboratory diagnosis methods (RT-PCR and antibody test) have to important drawbacks: (1) a low viral load conducts to a low detection rate, which can lead to false-negative results, and (2) the viral tests show positive/negative results but cannot judge the COVID-19 evolution in the chest [5]. In contrast, CT imaging and chest X-ray can be used to detect and measure the severity of the virus. In China, CT is widely used as a first line investigation method in patients with COVID-19 [6], and is recommended as the basis method for COVID-19 diagnosis

[7]. However, CT practice implies a high demand for radiology departments, and most importantly a decontamination of the equipment to reduce the risk of cross-infection. Hence, reducing the availability and applicability of the method [8].

Chest-Xray is one of the most common diagnosis methods for lung disease due to its accessibility and rapid analysis. Moreover, CXR is not as expensive as CT and does not require a previous preparation of the patient [9]. Therefore, CXR analysis can be applied to help medical professionals diagnose COVID-19 in patients. However, imaging diagnosis does come with its own complications, such as the difficulty of accurately distinguishing COVID-19 pneumonia from other forms of pneumonias caused by the cytomegalovirus, adenovirus, influenza A virus, influenza B virus, MERS, and other viral and bacterial pneumonias [2]. Consequently, this can lead to a COVID-19 diagnosis delay.

According to the Radiological Society of North America (RSNA), there are some characteristic manifestations on the CXR of a patient with Coronavirus [10]. These manifestations include the consolidation in the peripheral and mid to lower zone distribution, and the presence of bilateral patchy, bandlike ground-glass opacity [11]. In Fig. 1, we present two examples of CXR images. In (a) the image of a healthy patient is shown, in (b) an image from a person infected with COVID-19 is presented, and in (c) an image from a person with pneumonia [12].



*Figure 1.* Examples of CXR images taken from [12] a) Healthy patient. b) Patient infected with COVID-19. c) Patient with pneumonia.

Despite the similarities between different types of pneumonia, it is possible to optimize the COVID-19 diagnosis through machine-learning (ML) techniques. Over the years, ML methods have shown reliable results in the analysis of medical images [13]. Specially, deep learning architectures, like convolutional neural networks (CNNs), have demonstrated to achieve state-of-the-art performance in medical image classification [14, 15] and segmentation [16, 17, 18] tasks. As Gu et al. mentions, automated methods for medical diagnosis can play an important role for future diagnostic procedures, especially with the exploration of new CNN architectures that can improve the algorithm performance [19]. In the COVID-19 context, CNNs can recognize visual patterns from CXR images of COVID-19 patients and aid in the diagnosis, providing a rapid response and relieving the demand for radiology experts.

CNNs have obtained remarkable success by automatically optimizing its millions of parameters using labeled training data. Nevertheless, one of the biggest challenges is that such large-capacity trainers are prone to overfitting the training dataset if a small dataset is provided.

This is the case of most medical imaging problems, where acquiring labelled data is very expensive, time-consuming, and requires the participation of an expert radiologist. Furthermore, given the recent COVID-19 outbreak, getting sufficient images can be impossible. Data augmentation is a popular technique that has been presented to extend a dataset artificially. While different data augmentation strategies have been researched and applied in medical image classification [20], there has been little work on which operations work well in COVID-19 recognition which is a pivotal step in obtaining an effective and discriminative deep learning model.

In this paper, we apply a residual neural network (ResNet) [21] for COVID-19 automated diagnosis. The implemented ResNet can classify a patient's Chest-Xray image into COVID-19 positive, pneumonia caused from another virus or bacteria, and healthy. Furthermore, to increase the accuracy of the model and overcome the data scarcity of COVID-19 images, a personalized data augmentation strategy is used to enrich the dataset during the training process. Specifically, a three-step Bayesian hyperparameter optimization approach is applied to optimize nine hyperparameters that define the magnitude of the data augmentation operations and the model. Furthermore, we provide an insight into which data augmentation operations are successful in increasing a CNNs performance when doing medical image classification with COVID-19 CXR. In conclusion, the contributions of our work are as follows:

- We present the COVID-19 ResNet and its corresponding data augmentation strategy that achieves a competitive classification accuracy for COVID-19 detection.

- We provide an analysis of which data augmentation operations favor the classification accuracy in medical image classification, which will help researchers determine which operations to apply in their own models and specific problems.

## TOPIC DEVELOPMENT

### Literature Review

Medical imaging over the years has been essential for the visual representation of tissues and organs. Many imaging formats have been created such as Magnetic Resonance Imaging (MRI), X-ray, Computed Tomography (CT), in others [22]. The evaluation of a medical image is usually a manual and costly process, taking considerable time for a radiologist or medical expert to inspect all the slices and imaging modalities. A successful approach to shorten the inspection and evaluation time is to develop automatic models that extract the most important features from the images [23].

The first COVID-19 case appeared in December 2019, and until now it still presents a significant challenge to many countries in the world. To facilitate the diagnosis of COVID-19, researchers have focused on developing machine learning (ML) and deep learning (DL) methods that accelerate the detection of the virus in a patient's medical images. This is especially necessary since identifying the presence of the virus can help to reduce the transmission to uninfected individuals.

Various methods have been presented to automatically identify the disease using chest X-ray images, being CNNs the most used because of its high accuracy and precision. Wang et al. proposed a two-part classification method by applying a pre-trained inception network to convert the image data into one-dimensional feature vectors, followed by a fully connected network to produce the classification prediction. The study reports an accuracy of 79.3% on an external testing dataset [24]. In another study, Sethy et al. uses a two-step method to classify X-ray images.

First, they apply a deep learning architecture in the first layers to extract deep features from the image, and then apply a support vector machine in the last layer to perform the classification.

Using a ResNet50 as the deep learning architecture, they report an accuracy of 95.38% on a public dataset [25]. Xin et al. proposed an evolutionary multi-objective neural architecture search method, also known as EMARS-A, to automatically find the architecture of a CNN for COVID-19 classification. Their network achieves an accuracy of 89.67% [26]. On the other hand, Narin et al. developed a model that used five pre-trained convolutional neural networks based on the ResNet and Inception models and test them in three different binary datasets. An accuracy of 96.1% is obtained in dataset 1, 99.5% in dataset 2, and 99.7% in dataset 3 [27].

## **Methodology**

In this paper, we employ a convolutional neural network to distinguish CXR images from patients with COVID-19, patients with pneumonia caused by other infections, and healthy individuals. A Residual Convolutional Neural Network (ResNet) [21] is selected because of its capability to extract high level and complex features, which is necessary for the complicated task of medical image recognition. Moreover, to improve the classification accuracy and sensitivity of the model, the technique of data augmentation is extensively applied during training to increase the number of training samples, reduce overfitting, and increase the generalization capability of the model. To take the most advantage of the data augmentation method, the parameters are selected using a Bayesian hyperparameter optimization approach. In the following subsections we present the residual neural network applied, describe the data augmentation method, and finalize by presenting the Bayesian hyperparameter optimization approach used.

## **Residual neural network**

Neural Network depth has a strong influence in the accuracy of a network. As a network becomes deeper, it is more capable of recognizing and modelling the complex intricacies of the



images. However, when deep networks are being trained, it is difficult to propagate gradients specially to the deeper layers. Hence, giving rise to the gradient degradation problem. Gradient degradation causes information loss and reduces the accuracy of the model [21]. The residual network (ResNet) is a very deep network that implements a residual function to counteract the degradation problem. The residual function, also known as residual connection, is implemented through a summation function as shown in equation 1:

$$f(x) + x \quad (1)$$

Where the network layers are represented by  $f(x)$ , and  $x$  is the input feature map to the first layer. The residual connection adds a connection between the input to the layer and the output of the stacked layers. Thus, allowing the information to flow directly between network layers during the forward propagation, and most importantly permitting the gradients to pass during backpropagation. In Fig. 2, the basic residual block implemented in the COVID-19 ResNet classification architecture is presented.

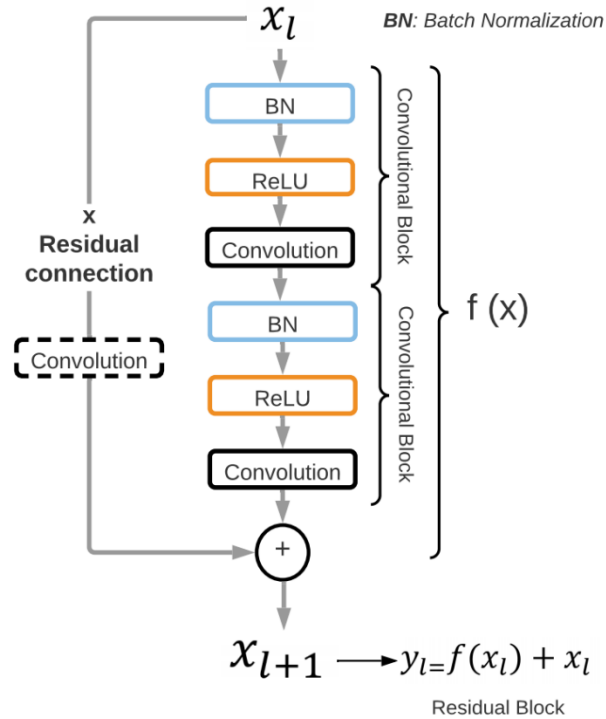
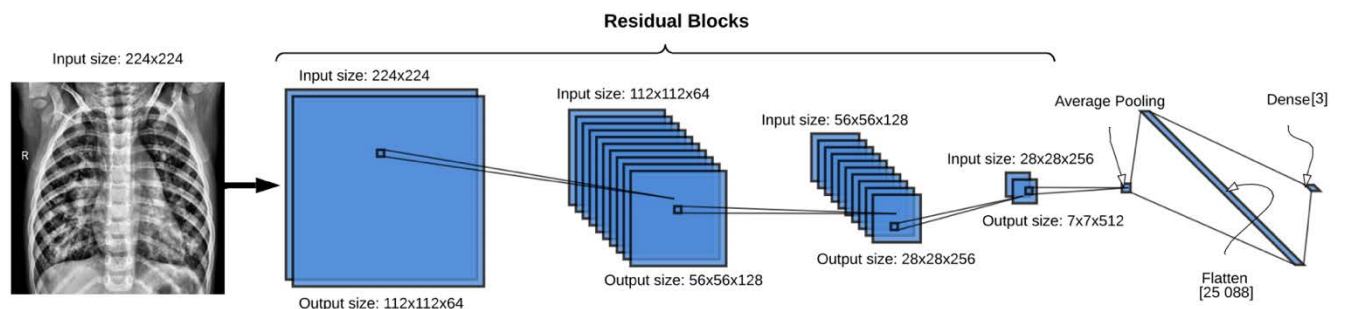


Figure 2. Basic residual block for the COVID-19 ResNet architecture. Each residual block  $y_l$  has a residual connection that adds the input image/input feature  $x_l$  of layer  $l$  to the output of the two convolutional blocks ( $f(x)_l$ ) producing the new input  $f$

The residual block, denoted as  $y_l$ , is composed of two convolutional blocks that represent  $f(x_l)$ . Each convolutional block is comprised of a batch normalization layer (BN), a ReLU activation function, and a convolutional layer. A ReLU activation function is implemented because it has shown to solve the vanishing gradient problem. Moreover, a batch normalization layer is also included to reduce the internal covariance shift and to normalize the output of each layer. The size of the convolutional kernels is a hyperparameter that is optimized with the Bayesian Optimization. The input feature  $x_l$  to layer  $l$  forms a residual connection with the transformed input  $f(x_l)$  through a summation operation, forming the input  $x_{l+1}$  to layer  $l + 1$  ( $x_{l+1} = y_l = f(x_l) + x_l$ ). Moreover, an extra convolutional layer is included before the residual connection

in residual blocks where the output features have a different size than the input features of next residual block. Hence, depending on the location of the residual block in the network, it might have two or three convolutional layers to guarantee the correct flow of information.

The entire ResNet architecture implemented is presented in Fig. 3. The input to the network is a 2D CXR image with shape  $224 \times 224 \times 1$ . 32 convolutional layers divided into 14 residual blocks make up the body of the network. The number of filters in the residual blocks increase progressively from 64 to 512. In detail, the ResNet structure has two residual blocks with 64 filters, 10 residual blocks with 128 filters, 10 residual blocks with 256 filters, and 2 residual blocks with 512 filters. On the other hand, as the number of filters is increased, the size of the feature maps is reduced by halved. In Fig. 3, the residual blocks grouped according to their number of filters and dimensions are presented. In the last layers, the ResNet applies an average pooling followed by a flatten layer to compress the features maps into a vector of size 25,088. Finally, a fully connected layer with 3 neurons and softmax classifier is used to predict the probabilities of an image being part of each of the three classes (COVID-19 positive, pneumonia caused by other bacteria or virus, healthy). Our network has a total of 15,676,549 trainable parameters which are optimized using the stochastic gradient descent method.



*Figure 3.* The COVID-19 ResNet architecture. Each blue square represents a residual block. The size of the input to the residual block is located on top of the residual blocks, and the output size on the bottom. Each residual block contains two convolutional blocs.

### **Data augmentation**

Neural network models can be quite successful when a huge amount of data is available. Nevertheless, in the medical field acquiring data is costly and sometimes unavailable, prohibiting the quantity of images obtained. Hence, CNNs tend to overfit the training dataset due to the considerable amount of model parameters that need to be fitted. This issue is aggravated by the fact that the COVID-19 outbreak is recent, so the number of publicly available images are limited.

Data augmentation is a method that increases the diversity of the training set by extending it artificially through the application of affine and elastic random transformation to the original images. The most common data augmentation operations are image rotation, reflection, horizontal and vertical shifting, color adjustments, and scaling. In this work, the data augmentation operations tested for training the COVID-19 ResNet are rotation, scaling, width shift, height shift, vertical flip, and horizontal flip because they have shown to increase the accuracy of neural networks. A description of the tested data augmentations are presented next, and an example of these transformations in a CXR image are shown in Fig. 4.

Rotation: Is an affine transformation that rotates an image  $I$  by an  $\theta$  angle around the center pixel. It is applied through the matrix  $R = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$ .

Scaling: Scales an image  $I$  in the  $x$  or  $y$  direction by applying the affine transformation  $S = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix}$ , where  $s_x$  and  $s_y$  are the scaling factors in  $x$  and  $y$  respectively. This operation helps the neural network learn from different shapes and sizes of the region of interest [28].

Width and Height shift: Shifts the image  $I$  by a given number of pixels in the horizontal or vertical axis. Since the area of interest can be shifted to different regions in the image, it forces the neural network to learn spatially invariant features.

Horizontal and Vertical Flip: This operation creates a reflection of the original image  $I$  along a selected axis. A flip in the horizontal axis swaps the right and left hemisphere. A vertical flip swaps the up and down section of the image. In natural images, only horizontal flips create real images. However, in medical images vertical flips have also shown to produce realistic images [28].

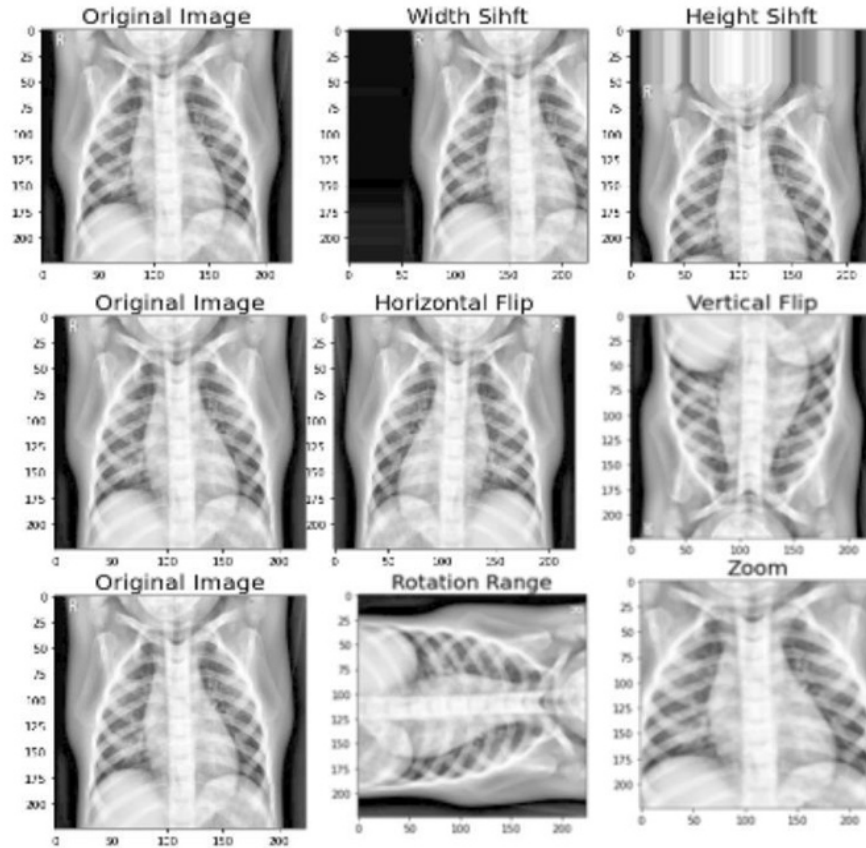


Figure 4. Data augmentation operations applied to a CXR image in the training dataset.

#### **Bayesian hyperparameter optimization for data augmentation.**

Determining the data augmentation operations that need to be applied in our specific problem, and to what extent, is a difficult task because there are many possible combinations that can be tested, and they affect the generalization and accuracy of the model. The correct value of these operations could be set manually by considering the values found in other publications [29]. However, data augmentation strategies successfully applied in a dataset may not transfer as effectively to another dataset due to particularities of each dataset and model. Hence, the data augmentation operations explained in section 3.2 (rotation, scaling, width shift, height shift, vertical flip, and horizontal flip) will be treated as hyperparameters in our model and its optimal

values selected using a Bayesian hyperparameter optimization approach. The search ranges for these hyperparameters are initially set to the maximum allowable value so to try to cover the whole search space.

Furthermore, four more hyperparameters related to the model and training process are added to the hyperparameter search. These hyperparameters are the learning rate, batch size, number of training epochs, and kernel size. Setting the correct learning rate is critical because a rate that is too large will converge very fast to a suboptimal solution, and if this value is too small the training process can get halted. There is no rule of thumb for this hyperparameter, it all depends on the specific neural network architecture. However, Bengio mentions a good initial learning rate is less than 1 and greater than  $1e-6$  [30].

To mitigate overfitting and underfitting, a good number of training epochs must be chosen. The number of epochs depends on the size of the dataset. Given that the dataset used in this work has a limited number of images, the range of the number of training epochs is set from 200 to 300.

Masters et al. recommends mini-batch sizes as small as 2 or 4 to improve network accuracy [31]. Considering these recommendations and the computational limitations, the range of the batch size is set between 1 and 3. All the hyperparameters being optimized are shown in the Table 1.

In the hyperparameter optimization problem, the objective is to find the hyperparameter values that minimize the validation cross-entropy loss function. Hence, the nine hyperparameters presented represent the decision variables, and their search range the search space. A Bayesian

optimization approach [32] is applied to solve this problem because it has proven to be effective in solving non-linear and non-convex optimization, and it reduces the search time through the application of Gaussian Process surrogate function.

The hyperparameters are optimized applying a three-step optimization process, in which the hyperparameter search bounds are adjusted progressively to reduce the algorithm search space and focus in the most promising search areas. Specifically, in each step we apply the Bayesian optimization approach, with a Gaussian Process surrogate function and Expected Improvement acquisition function, in the defined search bounds and find the best current solution. In the next step, we redefine the bounds by making them tighter to the previously found solutions. The hyperparameters and their respective search bounds in each step are presented in Table 1. 20 initial points were collected using a Latin hypercube sampling to approximate the Gaussian Process before running each optimization. The termination criteria for the optimization are 150 iterations or a distance of  $1e^{-8}$  or less between two consecutive points. The Bayesian optimization process is implemented using the GPyOpt library [33].

<b>Hyperparameters</b>	<b>Step 1</b>	<b>Step 2</b>	<b>Step 3</b>
<b>Rotation</b>	0 degrees – 359 degrees	0 degrees – 90 degrees	68.28 degrees – 240.41 degrees
<b>Zoom</b>	-1.0 – 1	0.0 – 0.5	0.34 – 0.37
<b>Width shift</b>	-1.0 - 1	0.0 - 0.5	0.19 – 0.35
<b>Height shift</b>	-1.0 – 1	0.0 – 0.5	-0.13 – 0.15
<b>Horizontal flip</b>	True (1) – False (0)	True (1) – False (0)	True (1) – False (0)
<b>Vertical flip</b>	True (1) - False (0)	True (1) - False (0)	True (1) - False (0)
<b>Learning rate</b>	$3e-4$ – $1e-5$	$3e-4$ – $1e-5$	$3e-4$ – $1e-5$
<b>Batch size</b>	1,2 and 3	1,2 and 3	1,2 and 3
<b>Kernel</b>	1,3 and 5	1,3 and 5	1,3 and 5
<b>Epochs</b>	200-300	200-300	200-250
<b>Minimized Loss</b>	0.2229209989309311	0.1504152715206146	0.1066201105713844

Table 1. Hyperparameters and search bounds optimized in each of the three steps of the Bayesian

optimization for the COVID-19 ResNet.



Finally, using the Bayesian Optimization approach to obtain the best hyperparameters requires high computational resources. Given our computational limitations, we apply a principal component analysis (PCA) to reduce the dimensionality of the data. 140 principal components per image are selected, which capture the 95% of the total variance of the originally 50,176-dimensional image. Therefore, the hyperparameter optimization is performed with the 140-dimensional reconstructed images. Applying the PCA reduction decreases drastically the RAM requirements from 15.51GB, when the original images are used, to 2.79 GB when the reconstructed images are applied. Despite PCA being an excellent approximator, using the reconstructed images can miss valuable information about the original input. In consequence, we only use these reconstructed images for performing the hyperparameter optimization and apply the full dimensional input for the model training and testing.

## **Experimental Setup**

### **Dataset.**

This study is performed in a publicly available dataset [12] composed of 420 2D CXR images in the Posterioranterior (P.A.) chest views. The images are classified in to three categories: COVID-19 patients (140 images), healthy or normal individuals (140 images), and patients with Pneumonia caused by bacteria or a virus different from COVID-19 (140 images). The dataset is available in the GitHub repository of the authors (<https://github.com/abzargar/COVID-Classifier.git>).

The dataset is split into 70% images for training, 15% images for validation and 15% images for testing. A simple random sampling method is applied to select the images for each set,

making sure that each category is equally presented in each set. The number of images per category and set are presented in Table 2.

Set	Category		
	Covid-19	Normal	Pneumonia
Training	98	98	98
Validation	21	21	21
Test	21	21	21
<b>Total</b>	140	140	140

Table 2. Number of CXR images per category for training, validation, and testing.

### Preprocessing operations.

The main motivations for image preprocessing is to standardize and enhance the image to facilitate the feature extraction. The preprocessing steps applied to the images are gray scale conversion, reseizing, and adaptive histogram equalization. An original image and its preprocessed counterpart are shown in Fig. 5. The preprocessing steps are explained next.

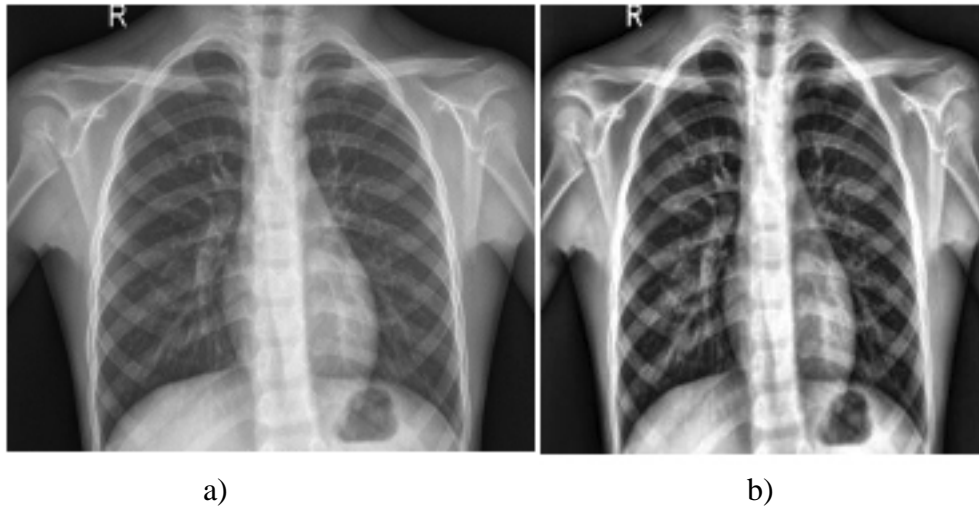


Figure 5. Examples of a preprocessed CXR image a) Non processed Normal CXR. b) Processed Normal 224x224 CXR

### **Gray scale conversion.**

Chest Xray-images contain considerable noise, such as blurring, fog, low contrast, and unwanted information. Gray scale conversion is a widely used technique to reduce the noise and computational cost of an image [34, 35]. This conversion transforms the color values of the original images (24 bit) represented in three dimensions XYZ (lightness, chroma and hue) into grayscale images represented only by the luminance (8 bit). The processed grayscale image has pixels values in the range of 0 (black) to 255 (white).

### **Resizing.**

The size of the images in the original dataset varies, so to prevent compilation errors during the pre-processing operations and model training, they are resized to a fixed 224x224 pixel size.

### **Adaptative histogram equalization.**

Histogram equalization is a contrast enhancement method that applies to each pixel in the image a mapping based on the surrounding pixels. A study made by Sherrier et al. shows that adaptative histogram equalization applied to chest radiography allow certain regions of the CXR to be enhanced differentially [36].

### **Training.**

The ResNet is trained with a cross-entropy loss function and the Adam optimizer with the parameters recommended in [37]. The weights are initialized from a gaussian distribution centered on 0 with a standard deviation of  $\sqrt{\frac{2}{f_{in}+f_{out}}}$ , where  $f_{in}$  are the number of input units in the weight tensor, and  $f_{out}$  the number of output units in the weight tensor. The number of training epochs are selected based on the results of the Bayesian optimization, as well as the data augmentation operations and its values. This information will be presented in the next section.

The code was developed using Google Collaboratory professional, with a high-RAM hosted runtime with 27.4 gigabytes available and 147.16 gigabytes of available disk space. The programming language used was Python V3.0 and the neural network implemented with the Keras library.

## RESULTS

### Bayesian optimization results.

The best hyperparameter values found in each step of the optimization process, their respective loss and validation accuracy are presented in Table 3. In the first step, the Bayesian optimization took 15.2 hours to find the best hyperparameters, reached an 0.22 loss, and a 90% validation accuracy. The second step took 11.19 hours, achieved a 92% validation accuracy, and a 0.15 loss. In the last step, the optimization converged in 5.4 hours, reached the highest validation accuracy of 97%, and a loss of 0.10. The hyperparameters found in the third step are the values used for the COVID-19 ResNet training and testing.

Furthermore, to test the adequacy of the Bayesian Optimization, we train the ResNet architecture using data augmentation values recommended by literature [38]. The hyperparameter values tested are presented in Table 4. A 95% validation accuracy is reached, which is 2% lesser than the accuracy obtained using the three-step optimization approach. Considering the need to accurately diagnose COVID-19 in a patient a 2% increase is an important gain.

Hyperparameters	Step 1	Step 2	Step 3
<b>Rotation</b>	240.41 degrees	64.28	187.5 degrees
<b>Zoom</b>	0.34	0.37	0.36
<b>Width shift</b>	0.36	0.19	0.23
<b>Height shift</b>	-0.13	0.16	-0.13
<b>Horizontal flip</b>	True (1)	True (1)	True (1)
<b>Vertical flip</b>	False (0)	True (1)	False (0)

<b>Learning rate</b>	0.0003	0.00025	2e-4
<b>Batch size</b>	3	2	3
<b>Kernel</b>	1	3	1
<b>Epochs</b>	200	300	200
<b>Minimized Loss</b>	0.223	0.150	0.107
<b>Validation Accuracy</b>	90%	92%	97%

*Table 3.* Best hyperparameter values found in each step of the three-step Bayesian hyperparameter optimization. The best hyperparameters are the ones found in step 3.

<b>Hyperparameters</b>	<b>Experiment 3</b>
<b>Rotation</b>	45
<b>Zoom</b>	0.2
<b>Width shift</b>	0.2
<b>Height shift</b>	0.2
<b>Horizontal flip</b>	True (1)
<b>Vertical flip</b>	True (1)
<b>Learning rate</b>	1e-6 - 1
<b>Batch size</b>	2 or 4
<b>Kernel</b>	3
<b>Epochs</b>	200
<b>Validation Accuracy</b>	95%

*Table 4.* Hyperparameters tested based on literature.

### **Classification results.**

In this subsection we present the classification results obtained by the best ResNet (the ResNet trained with the optimal hyperparameter values obtained in 4.4.1) in the test set. The metrics used for the evaluation are the accuracy, precision, recall, and f1-score. The metrics are defined in equations 2-5.

$$Accuracy = \frac{TN+TP}{TN+TP+FN+FP} \quad (2)$$

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

$$Recall = \frac{TN+TP}{TN+TP+FN+FP} \quad (4)$$

$$F1 = \frac{Precision*Recall}{Precision+Recall} * 2 \quad (5)$$

The evaluation metrics in the test set are presented in the Table 5. The accuracy reached by the model is of 94%, with 59 correctly classified samples out of 63. In terms of the precision, the model correctly detects the positive COVID-19 and Pneumonia cases with a high 95%. Meanwhile, the normal cases have a lower positive detection of 91%. Analyzing the recall, in the normal class a 100% recall is obtained, while for the COVID-19 class a 95%, and for the pneumonia class a 86%. Recall aims to minimize the false negatives, hence having a 100% recall in the normal class means the model perfectly distinguishes patients with pneumonia caused by COVID-19 or another virus and bacteria from healthy individuals. Finally, the f1-score provides as harmonic mean between the precision and recall. All classes have an f1-score more than or equal to 90%, which means that the model provides an adequate balance between precision and recall.

<b>Classification Results</b>			
<b>Category</b>	<b>Precision</b>	<b>Recall</b>	<b>f1-Score</b>
<b>Covid</b>	0.95	0.95	0.95
<b>Normal</b>	0.91	1.00	0.95
<b>Pneumonia</b>	0.95	0.86	0.90
<b>Metrics</b>			
<b>Accuracy</b>	0.94		

Table 5. Classification results for each category with the COVID-19 ResNet in the test set.

### Benchmark Comparison.

To compare the performance of our network, in Table 6 we present the results of our work and highly cited studies that perform COVID-19 image classification. Furthermore, to provide a better background of each study, the preprocessing techniques and architecture implemented by each work is included. As it can be seen, the ResNet is a popular architecture but what distinguishes ours is the application of the Bayesian optimization to find the best data augmentation and model hyperparameters. The accuracy of the proposed COVID-19 ResNet has a competitive accuracy, being ranked third against the other works. It is also ranked second in terms of the sensibility.

Model	Architecture	Preprocessing	Accuracy (%)	Recall (%)	F1-score (%)
Maghdid et al. [39]	AlexNet	Cropping, resizing	98	-	-
Farooq et al. [40]	ResNet50	Rescaling, data augmentation, normalizing	96.23	100	100
<b>COVID-19 ResNet (Ours)*</b>	<b>ResNet</b>	<b>Data Augmentation, normalizing, RGB, Color adaptative, reduce dimension</b>	<b>94</b>	<b>95</b>	<b>95</b>
Wang et al. [41]	COVID-Net	Rescaling, data augmentation	93.3	91	-
Wu et al. [42]	ResNet50	Segmentation, Rescaling, Multiview Fusion	76	81.1	-

Table 6. Evaluation metrics of the implemented COVID-19 ResNet (\*) and competing state-of-the-art models on COVID-19 image classification. A dash means the authors have not computed the metric.

## CONCLUSIONS

The aim of this work was to develop a CNN to automatically classify CXR images into COVID-19 positive, pneumonia caused by a bacteria or a virus other than COVID-19, and healthy. A ResNet architecture was implemented due to its capability to extract complex short- and long-range features, while preventing the gradient degradation problem. The network was trained in a publicly available dataset composed of 420 2D chest X-rays. To overcome the challenges of working with a small dataset, a data augmentation technique was applied. Six data augmentation operations were tested, namely rotation, scaling, width shift, height shift, horizontal flip, and vertical flip. Furthermore, the learning rate, kernel size and batch size of the model were also optimized for this problem. The best values for these nine hyperparameters were selected using a three-step Bayesian optimization approach.

In each step of the three-step optimization process, the Bayesian hyperparameter optimization algorithm minimized the validation cross-entropy loss while searching for the best hyperparameters values. In the initial step, the search bound for the hyperparameters was set to the maximum range allowed by the operation. In next steps, the bound was progressively adjusted around the best solutions found so far to reduce the feasible space and exploit the search region. The strategy showed to provide good results, as the best solutions found in each step kept increasing the validation accuracy and reducing the cross-entropy loss function. Moreover, meanwhile the ResNet with the optimized hyperparameters obtained a 97% validation accuracy, the ResNet with the recommended hyperparameter values found on the literature reached a 95% validation accuracy. Hence, this three-step optimization process improved in 2 percentual points the validation accuracy, which is an important gain considering the need to accurately diagnose a COVID-19 positive patient to reduce the transmission to uninfected individuals.



In reference to the data augmentation operations, the results obtained through the Bayesian optimization showed that using a vertical flip does not help increase the classification accuracy. This is an interesting finding, since vertical flips have shown to be useful in other types of medical image classification problems and used as default operation when training the networks. Horizontal flips, on the other hand, helped to increase accuracy and are highly recommended to be applied in COVID-19 recognition tasks. In reference to the rotation operation, it showed to be successful technique, but the angle of rotation should not go beyond 190 degrees as the accuracy starts to decrease. Scaling was also found to be a beneficial operation when a maximum of 37% zoom was applied. In general, scaling over a 40% can cause some of the regions of interest to be missed, hence reducing the recognition capability of the model.

Finally, the height and width shift did seem to help increase the accuracy but in small values. In general, all data augmentation operations should be included during training, with exception of the vertical flip, but the magnitude of the operation should be set to medium values (as shown in Table 2). If extreme values are used in data augmentation the artificially produced images do not adhere with reality and affect the training process of the network. Hence, the validation accuracy decreases instead of increasing. In conclusion, the values obtained in this work for the data augmentation operations may be highly applicable for another research focused on CXR chest classification where only a limited dataset is available.

Our Python codes and processed datasets are available on GitHub:

<https://github.com/titulacion2021/Image-Classification-ResNet>

## REFERENCES

- [1] World Health Organization. (2020). Retrieved february 20, 2021, from <https://www.who.int/emergencies/diseases/novel-coronavirus-2019/question-and-answers-hub/q-a-detail/coronavirus-disease-covid-19#:~:text=symptoms>
- [2] Corman, V., Landt, O., Kaiser, M., Molenkamp, R. M., Chu, D., & Drosten, C. (2020). Detection of 2019 novel coronavirus (2019-nCoV) by real-time RT-PCR. *Eurosurveillance*, 25(3), 2000045.
- [3] Center of Disease ,Control and Prevention. (2021). Retrieved february 19, 2021, from <https://www.cdc.gov/coronavirus/2019-ncov/testing/diagnostic-testing.html>
- [4] Center of Disease, C. a. (2021, february 2). Retrieved february 20, 2021, from <https://www.cdc.gov/coronavirus/2019-ncov/testing/serology-overview.html>
- [5] Dai, W. C., Zhang, H. W., Yu, J., Xu, H. J., Chen, H., Luo, S. P., & Lin, F. (2020). CT Imaging and Differential Diagnosis of COVID-19. *Canadian Association of Radiologist Journal*, 2(71), 195-200.
- [6] Wong, H., Lam, H., Fong, A., IEUNG, S., Chin, T., & Lo, C. (2020). Frequency and distribution of chest radiographic findings in patients positive for COVID-19. *Radiology*, 296(2), E72-E78.
- [7] Wan, Y., Shang, J., Graham, R., Baric, R., & Li, F. (2020). Receptor recognition by novel coronavirus from Wuhan: An analysis based on decadelong structural studies of Sars. *Journal of Virology*, 94(7).
- [8] Radiology, A. C. (n.d.). *Recomendations for the use of the Chest Radiography and Computed Tomography (CT) for suspected COVID-19 Infection*. Retrieved february 21, 2021, from <https://www.acr.org/Advocacy-and-Economics/ACR-Position-Statements/Recommendations-for-Chest-Radiography-and-CT-for-Suspected-COVID19>
- [9] MDPI. (2020, August 26). Retrieved February 21, 2021, from <https://www.mdpi.com/2227-7390/8/9/1423>
- [10] RSNA. (2020, Abril 15). Retrieved February 20, 2021, from [https://pubs.rsna.org/doi/10.1148/radiol.2020201393?\\_ga=2.72325459.1392118184.1617117795-543966827.1617117790](https://pubs.rsna.org/doi/10.1148/radiol.2020201393?_ga=2.72325459.1392118184.1617117795-543966827.1617117790)
- [11] Smith, D., Grenier, J., Batter, C., & Spieler, B. (2020). A Characteristic Chest Radiographic Patter in the Setting of the COVID-19. *Radiology: Cardiothoracic Imaging*, 2(5), e200280.
- [12] Khuzani, A. Z., Heidari, M., & Shariati, S. A. (2020, May 8). COVID-Classifer: An automated machine learning model to assist in the diagnosis of COVID-19 infection in chest x-ray images. *medRxiv*. Retrieved March 6, 2021, from [https://github.com/abzargar/COVID-Classifer/tree/master/dataset/original\\_images\\_preprocessed](https://github.com/abzargar/COVID-Classifer/tree/master/dataset/original_images_preprocessed)
- [13] Dai, W.-c., Zhang, H.-w., Yu, J., Xu, H.-j., Chen, H., Luo, S.-p., . . . others. (2020). CT Imaging and Differential Diagnosis of Covid-19. *Canadian Association of Radiologists Journal*, 71(2), 195-200.
- [14] Parvathy, V. S., Pothiraj, S., & Sampson, J. (2020). Optimal Deep Neural Network model based multimodality fused medical image classification. *Physical Communication*, 41.
- [15] Godasu, R., Zeng, D., & Sutrave, K. (2020). Transfer learning in medical image classification: Challenges and opportunities. *Transfer*, 5, 28-2020.
- [16] Baldeon-Calisto, M., & Lai-Yuen, S. K. (2020). AdaResU-Net: Multiobjective adaptive convolutional neural network for medical image segmentation. *Neurocomputing*, 392, 325-340.

- [17] Calisto, M. B., & Lai-Yuen, S. K. (2020). AdaEn-Net: An ensemble of adaptive 2D-3D Fully Convolutional Networks for medical image segmentation. *Neural Networks*, 126, 76-94.
- [18] Calisto, M. B., & Lai-Yuen, S. (2021). EMONAS: efficient multiobjective neural architecture search framework for 3D medical image segmentation. *Medical Imaging 2021: Image Processing*.
- [19] Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., & Chen, T. (2018). Recent avances in convolutional neural networks. *Pattern Recognition*, 77, 354-377.
- [20] Hussain, Z., Gimenez, F., Yi, D., & Rubin, D. (2017). Differential data augmentation techniques for medical imaging classification tasks. *AMIA Annual Symposium Proceedings*.
- [21] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- [22] Aggarwal, P., Vig, R., Bhadhoria, S., & Dethe, C. (2011). Role of segmentation in medical imaging. *International Journal of Computer Applications*, 975(8887), 54-61.
- [23] Müller, D., & Kramer, F. (2021). MIScnn: a framework for medical image segmentation with convolutional neural networks and deep learning. *BMC medical imaging*, 1-2.
- [24] Wang, S., Kang, B., Ma, J., Zeng, X., Xiao, M., Guo, J., . . . Meng, X. (2021). A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19). *European Radiology*, 1-9.
- [25] Sethy, P. K., Behera, S. K., Ratha, P. K., & Biswas, P. (2020). *Detection of coronavirus disease (covid-19) Based on Deep Features*. PrePrints.
- [26] He, X., Shihao, W., Guohao, Y., Jiyong, Z., & Chu, X. (2020). Efficient Multi-objective Evolutionary 3D Neural Architecture Search for COVID-19 Detection with Chest CT Scans. *arXiv preprint arXiv:2101.10667*, (pp. 1-2).
- [27] Narin, A., Kaya, C., & Pamuk, Z. (2020). Automatic Detection of Coronavirus Disease (COVID-19) Using X-ray Images and Deep Convolutional Neural Networks. *arXiv preprint arXiv:2003.10849*, (pp. 2-3).
- [28] Zeshan, H., Gimenez, F., Yi, D., & Rubin, D. (2018). Differential Data Augmentation Techniques for Medical Imaging Classification Tasks. *Annual Symposium*. San Francisco.
- [29] Bergstra, J., Barnedet, R., Bengio, Y., & Kégl, B. (2011). Algorithms for hyper-parameter optimization. *In Advances in Neural Information Processing Systems (NIPS)*, (pp. 2546-2554).
- [30] Bengio, Y. (2012). Practical Recommendations for Gradient-Based Training of Deep Architectures. *Neural Networks: Tricks of the trade*, 437-478.
- [31] Masters, D., & Carlo, L. (2018). Revisiting small batch training for deep neural networks. *arXiv preprint arXiv:1804.07612*, (pp. 2-3).
- [32] Snoek, J., Rippel, O., Swersky, K., Kiros, R., Satish, N., Sundaram, N., . . . Adams, R. (2015). Scalable bayesian optimization using deep neural networks. *International conference on machine learning*.
- [33] The GPyOpt authors. (2016). GPyOpt: A Bayesian Optimization framework in Python. <http://github.com/SheffieldML/GPyOpt>.
- [34] Kaur, S., & Kaur, S. (2014). *An Efficient Approach for Number Plate Extraction from Vehicles Image under Image Processing Article*. Jalandhar: CT Group of Institutions.

- [35] Macêdo, S., Givânio, M., & Kelner, J. (2015). *A comparative study of grayscale conversion techniques applied to SIFT descriptors*. Brazil: Universidade Federal de Pernambuco.
- [36] Sherrier, R. H., & Jonhson , G. A. (1987). Regionally Adaptative Histogram Equalization of the Chest. *IEEE Transaction on Medical Imaging*, 6(1), 1-7.
- [37] Kingma, P. D., & Ba, L. J. (2015). ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION. *ICLR*.
- [38] Gu, S., Pednekar, M., & Slater, R. (2019). Improve Image Classification Using Data Augmentation and Neural Networks. *SMU Data Science Review*, 2, 1.
- [39] Maghdid, H., Assad, K., Ghafoor, A., & Khan, M. (2020). Diagnosing covid-19 pneumonia from x-ray and ct images using deep learning and transfer learning algorithms. *arXiv*, 00038(2004).
- [40] Farooq, M., & Hafeez, A. (2020). Covid-resnet: A deep learning framework for screening of covid19 from radiographs. *arXiv*, 2003(14395).
- [41] Wang, L., Lin, Z. Q., & Wong, A. (2020). Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. *Scientific Reports*, 10(09871), 1-12.
- [42] Wu, X., Hui, H., Niu, M., Li, L., Wang, B., Li, H., & Tian, J. (2020). Deep learning-based multi-view fusion model for screening 2019 novel coronavirus pneumonia: a multicentre study,. *European Journal of Radiology*, 128, 109041.