T.C.

ALTINBAS UNIVERSITY

Institute of Graduate Studies

Information Technologies

# DETECTION OF COVID-19 PNEUMONIA EFFECTS IN CHEST X-RAYS USING DEEP LEARNING

Mohammed Khamees AHMED

Master of Science

Supervisor

Asst. Prof. Dr. Ayça Kurnaz TÜRKBEN

Istanbul, 2021

# DETECTION OF COVID-19 PNEUMONIA EFFECTS IN CHEST X-RAYS USING DEEP LEARNING

by

**Mohammed Khamees AHMED**

Information Technologies

Submitted to the Institute of Graduate Studies

in partial fulfillment of the requirements for the degree of

Master of Science

ALTINBAŞ UNIVERSITY

2021

The thesis titled “DETECTION OF COVID-19 PNEUMONIA EFFECTS IN CHEST X-RAYS USING DEEP LEARNING” prepared and presented by “Mohammed Khamees AHMED” was accepted as a Master of Science Thesis in Information Technologies.

______________________________

Asst. Prof. Ayça Kurnaz TÜRKBEN

Supervisor

Thesis Defense Jury Members:

| | | |
|---|---|---|
| Asst. Prof. Dr. Ayça Kurnaz TURKBEN | School of Engineering and Architecture, Altinbas University | __________________ |
| Asst. Prof. Dr. Abdullahi Abdu IBRAHIM | School of Engineering and Architecture, Altinbas University | __________________ |
| Asst. Prof. Dr. Tareq Abed MOHAMMED | Computer Science and Information Technology, Imam Ja'afar AL-Sadiq University | __________________ |

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Approval Date of Institute of Graduate Studies: ____/____/____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Mohammed Khamees AHMED

# DEDICATION

First and foremost, I would like to thank Allah Almighty for giving me the knowledge, ability, and opportunity to undertake this research study and to persevere and complete it satisfactorily. Heartfelt thanks go to my father and my mother. Every success is a direct consequence of their influence in my life and their love. Also, I don't forget to thank "Asst. Prof. Dr. Sefer KURNAZ" for his advice, In the end, I have to mention All my brothers and my friends for their support and love.

# ACKNOWLEDGEMENTS

All praises are to ALLAH for bestowing me with health, opportunity, patience, and knowledge to complete this research.

My profound gratitude goes to my Supervisor Asst. Prof. Dr. Ayça Kurnaz TÜRKBEN, for her invaluable guidance, excellent supervision, continuous encouragement, and constant support in making this research possible. Her cooperation, tolerance, constructive criticism, and useful suggestions have been of immense encouragement to me and enabled me to develop a deeper understanding of this research. I sincerely thank her for the time spent proofreading and correcting my mistakes.

I am especially indebted to my parents, who were my first teachers in this world by setting a good example for me about how to live, study, work, and for their love, sacrifices, and support.

I also acknowledge with thanks my wife's for her support, encouragement, and prayers.

I extend my gratitude to all those who were directly or indirectly involved by either encouraging, praying, and offering constructive advice in this thesis.

Thank you.

# ABSTRACT

## DETECTION OF COVID-19 PNEUMONIA EFFECTS IN CHEST X-RAYS USING DEEP LEARNING

Mohammed Khamees AHMED

M.Sc., Information Technologies, Altınbaş University,

Supervisor: Asst. Prof. Dr. Ayça Kurnaz TÜRKBEN

Date: 10/2021

Pages: 53

The development of technological tools based on artificial intelligence (AI) could contribute significantly in the fight against COVID-19. AI is the ability of a machine to apply human cognitive functions. In this paper we propose a deep learning based model for covid-19 detection relying on the effects it yields on the lungs.

**Keywords:** MRI, XRAYS, Covid-19, Deep Learning, SVM

# TABLE OF CONTENTS

**Pages**

**ABSTRACT** ........ **vii**

**LIST OF TABLES** ........ **ix**

**LIST OF FIGURES** ........ **x**

**LIST OF ABBREVIATIONS** ........ **xi**

**LIST OF SYMBOLS** ........ **xii**

**1. INTRODUCTION** ........ **1**

1.1 BACKGROUND ........ 1

1.2 PROBLEM STATEMENT ........ 2

1.3 MOTIVATION ........ 2

1.4 CONTRIBUTION ........ 3

1.5 THESIS ORGANIZATION ........ 4

**2. LITRITAURE REVIEW** ........ **5**

**3. MATERIALS AND METHODS** ........ **7**

3.1 COVID 19 AND MACHINE LEARNING ........ 8

3.2 DEEP LEARNING ........ 10

3.2.1 Convolutional Neural Network. ........ 11

3.2.2 SVM Classification ........ 13

3.3 IMAGE SEGMENTATION ........ 20

**4. PROPOSED METHOD** ........ **24**

**5. SIMULATION AND RESULT** ........ **32**

**6. CONCLUSION** ........ **38**

**REFERENCES** ........ **39**

**APPENDIX OF ALL THE CODES USED IN THIS THESIS** ........ **44**

# LIST OF TABLES

**Pages**

Table 5.1: Result ........................................................................................................ 33

Table 5.2: Result ........................................................................................................ 36

# LIST OF FIGURES

**Pages**


Figure 1.1: Covid-19 rise in cases ........................................ 1

Figure 1.2: Covid-19 damage to human lungs ........................................ 3

Figure 2.1: Shows the structure of an CNN ........................................ 5

Figure 3.1: Covid 19 and AI ........................................ 8

Figure 3.2: Effects of COVID19 in Xrays ........................................ 9

Figure 3.3: CNN AlexNet image net ........................................ 12

Figure 3.4: SVM hyperplame ........................................ 15

Figure 3.5: SVM classification using hyper parameters ........................................ 18

Figure 3.6: Lung X-rays segmentation ........................................ 21

Figure 4.1: Training model for the Covid-19 detector ........................................ 25

Figure 4.2: data entry form covid-19 ........................................ 26

Figure 4.3: X-ray model after modification ........................................ 27

Figure 4.4: Infected x-ray model covid-19 ........................................ 28

Figure 4.5: X-ray model not infected with covid-19 ........................................ 30

Figure 4.6: Types of viral infections ........................................ 31

Figure 5.1: Training Loss and Accuracy on COVID-19 Dataset ........................................ 32

Figure 5.2: Training and testing loss ........................................ 34

Figure 5.3: Training and testing accuracy ........................................ 35

# LIST OF ABBREVIATIONS

| | | |
|---|---|---|
| AI | : | Artificial Intelligence |
| RTPCR | : | Reverse-Transcription Polymerase Chain Reaction Test |
| CNN | : | Convolutional Neural Networks |
| SVM | : | Support Vectors Machine |
| KKT | : | Karush-Khun-Tucker |
| PACS | : | Image Archiving and Communication System |

# LIST OF SYMBOLS

$\sum$ : "sigma" Denotes the sum of a finite number of terms

$\geq$ : Angle frequency

$\in$ : Denotes set membership, and is read "in" or "belongs to"

# 1. INTRODUCTION

## 1.1 BACKGROUND

Currently the world is dealing with a pandemic of the novel coronavirus, severe acute respiratory syndrome coronavirus 2 or SARS-CoV-2, which has become known as COVID-19, having its first reported case in Wuhan, China, not the end of the year of 2019. Due to its rapid dissemination, it has become a serious public health problem in the world. In order to diagnose COVID-19, or test that you see the or RTPCR being used more due to its lack in some locations and a delay in obtaining the results, it becomes necessary to identify and develop ferments that can help you. A number of alternatives to aid in the identification of COVID-19 and the use of chest radiography, which shows characteristics similar to other pneumonia caused by other coronaviruses in the meantime, a rapid radiological interpretation of images is always available, particularly where we find few resources in that pneumonia (caused by viruses and bacteria) has a higher incidence and higher mortality rates.:

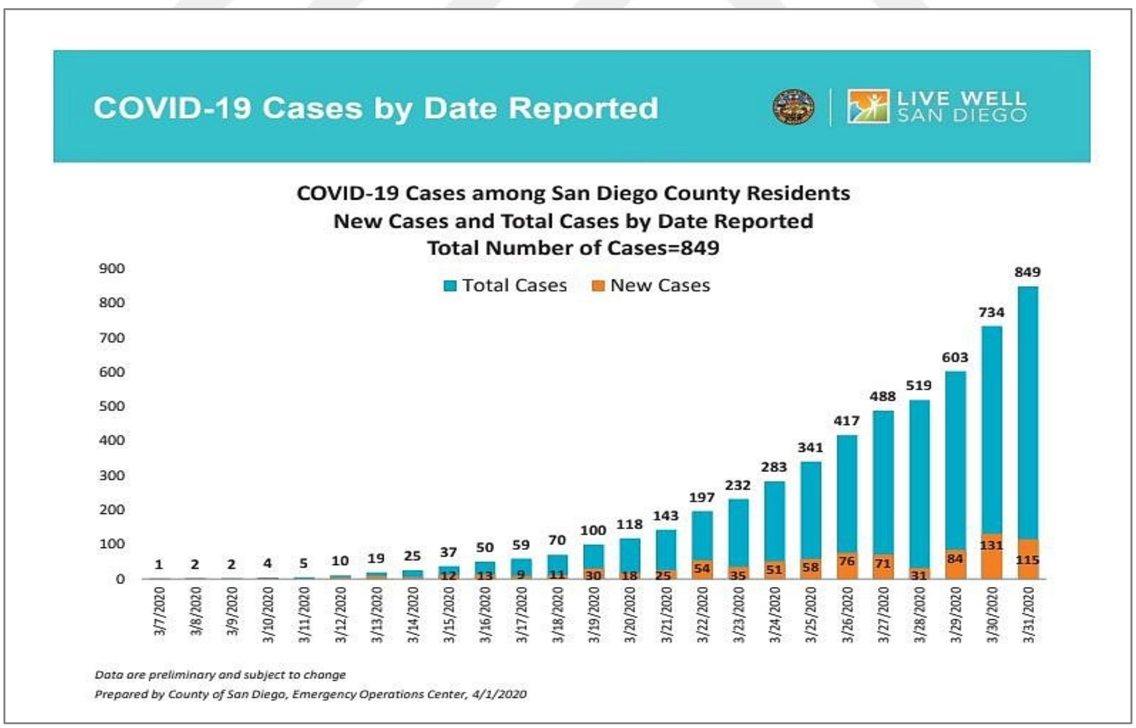



**Figure 1.1**: Covid-19 rise in cases.

## 1.2 PROBLEM STATEMENT

The process of interpreting agents (brain tumors or lung anomalies, for example) is a complex activity and, therefore, is necessary, or the use of image processing techniques, often combined with machine learning techniques, to identify -A application of deep learning techniques (deep learning) to classify images of x-ray images and a considerable growth in recent years. Several investigations address this topic, such as performing the classification of images to help prevent early diagnosis of tuberculosis, and classify injuries through chest radiography also used for the classification of x-ray images of the x-ray with pneumonia and other injuries.

## 1.3 MOTIVATION

The field of computer vision has seen tremendous progress in a wide range of applications due to the use of deep learning and especially the existence of large data sets of annotated images [1]. Significant improvements in the performance of previously considered difficult problems, such as object recognition, detection, and segmentation, were demonstrated via methods based on manually acquiring image properties [2]. In recent years, it has been proven that deep learning can continually improve and enable performance increases in all areas, especially in the field of medical image analysis and in object detection and segmentation tasks [3]. Recent work in the field of machine learning includes important medical applications, for example, in the field of pulmonology (classification of lung diseases [4] where deep learning can detect lung nodules on computed tomography images [5]). Compared to other computer vision applications, the main limitation in medical machine learning applications is that most of the methods that have been proposed are evaluated through the use of a fairly small data set with hundreds of patients, but in recent years some progress has been made through the introduction of publicly available data sets. In this article, we focus our efforts on identifying and classifying chest diseases, a problem that we present in the following sections...

**Figure 1.2**: Covid-19 damage to human lungs.

## 1.4 CONTRIBUTION

Currently the world is dealing with a pandemic related to a new coronavirus, severe acute respiratory syndrome coronavirus 2 or SARSCoV-2, which has become known as COVID-19. A number of alternatives to aid in the identification of COVID-19 and the use of chest radiography, which shows characteristics similar to other pneumonia caused by other coronaviruses. In the meantime, a rapid radiological interpretation of images is always available. An application of deep learning techniques (deep learning) to classify images of x-ray images and a considerable growth in recent years. In this work, we used two convolutional neural network architectures, InceptionResNetV2 and ResNetX50, as the objective of solving the problem of classifying X-ray images of people with pneumonia, in order to help not pre-diagnosis, mainly, of COVID-19, becoming a possible method of triage of patient's architecture obtain the best results for all defined metrics.

## 1.5 THESIS ORGANIZATION

The thesis is structured as follows: Section 2 is where we review some of the previous work and implementation on smart grids. Section 3 is where we give a background about all the components. Section 4 is where we explain our model in details and implementation of that model in details, Section 5 is where we simulate our model and record the results obtained. Section 6 is where we conclude our work and put a future scope into perspective.

# 2. LITRITAURE REVIEW

The deep learning paradigm (of English deep learning) is a sub-area of machine learning that addresses or automated learning, being carried out through the use of successive deep layers within a neural network architecture as explained in the works of (FRANCOIS, 2017). As a result, each litter passes or the result of its learning for the next litter. The greater the quantity of litters, the deeper the neural network becomes. A main characteristic of deep learning is the use of litters to perform the extraction of characteristics and classification of two dice for the works of (RIGHETTO, 2016). As deep learning algorithms become viable options for image classification tasks, for example. CNN is biologically inspired architectures capable of training and learning / generalizing invariant representations at scale, translation, rotation and related transformations (LIU; ROSENBERG; ROWLEY, 2007; JURASZEK et al., 2014) further showed that in their hypothesis. As CNNs composed of two types of algorithms of the area known as deep learning and are protected for common use of multiple dimensions, making them candidates for solving problems involving reconstructing of images of (AREL et al., 2010). For nearly six years traditional Computational Viewing processes, an CNN is capable of applying non-structured given filters, maintaining the relationship of visibility between the image pixels or the long process of the network.

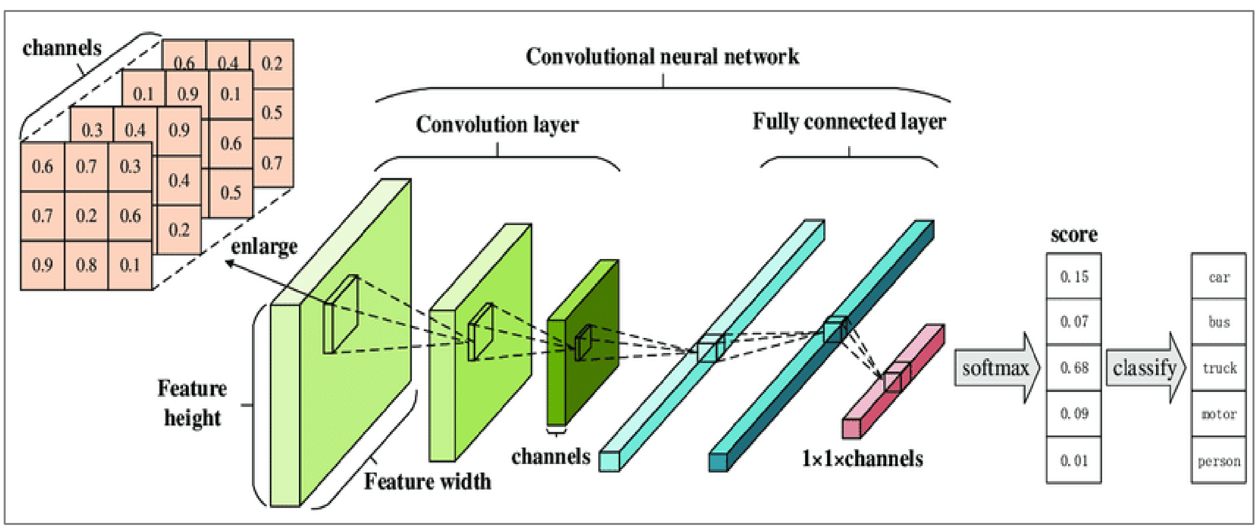



**Figure 2.1:** shows the structure of an CNN.

As CNNs have been used by the scientific community in medical images because of their excellent performance demonstrated in Computational Vision. Usually, as shown in Figure 2.1, three types of litters are used to build an CNN (MOUSSER; OUADFEL, 2019): convolutional litter, pooling litter and fully connected litter. A Python programming language was used for the implementation of CNNs, as an aid to the Keras library, widely used in the construction of deep neural networks, executing on a TensorFlow open code library (FOUNDATION, 2020; KERAS, 2020; TENSORFLOW, 2020). As architects undertaken in this study, I also study InceptionResNetV2, which has 572 litters; e to ResNeXt50, which has 50 litters. The three proposed architects were executed in a virtual machine (VM) provided by Google Collaboratory (Collab) (COLAB, 2020). O Collaborative objective to disseminate and encourage research in the machine learning area. O Colab makes available pre-configured VMs with machine learning and artificial intelligence essential libraries, such as TensorFlow, Matplotlib and Keras. To obtain two models, a data base with chest X-ray images was used, widely used without diagnosing pneumonia. Seus developers (CHOWDHURY et al., 2020) will use four databases: SIRM COVID-19 (INTERVENTISTICA, 2020), public database available on GitHub (COHEN; MORRISON; DAO, 2020), collection of images in different sources and Chest x-ray dice base available no Kaggle (KERMANY et al., 2018b).

# 3. MATERIALS AND METHODS

The development of technological tools based on artificial intelligence (AI) could contribute significantly in the fight against COVID-19. AI is the ability of a machine to apply human cognitive functions, such as machine learning, deep learning, natural language processing, computer vision, among others, to through training with data and computational predictions based on pattern recognition the information into five approaches that could help combat this pandemic: early detection, diagnosis, prognosis, use of the data panel, and treatment and control of COVID-19 The first approach taken was the use of AI for the early detection of epidemic outbreaks through a search analysis system for key terms on digital platforms, thanks to which it was possible to send the first global alert of this pandemic. The second approach was the use of deep learning algorithms to diagnose COVID-19, automatically evaluating chest tomography images with abnormalities characteristic of SARS-CoV-2 infection. The third approach was the prognosis of patients with COVID-19, in order to focus care and plan the use of medical resources only in critical patients; however, access to digitized clinical data and a trained prediction model was required. The fourth approach was the implementation of data panels to visualize the incidence of the pandemic in real time and to track the virus genome on a global platform where sequences and their mutations are shared. Finally, the fifth focus was the treatment and control of COVID-19; On the treatment side, with the use of computational search algorithms, among millions of molecular structures, possible drug candidates that could contribute to the development of a drug or vaccine to combat the pandemic were quickly identified; and on the control side, artificial vision technologies were implemented (infrared cameras to detect body temperature, facial recognition systems to verify the use of masks and compliance with social distancing), a QR code system was designed for monitoring of cases and, in its broadest sense, various fields of AI and engineering were used to create robots for cleaning, disinfection and medical care, in order to minimize contact between patients and health personnel. The use of AI to help fight the pandemic should be considered as support for non-pharmacological interventions. Some studies have shown that AI offers a diagnostic accuracy similar to that of professionals, being even faster and cheaper than standard tests (4). In contrast, the use of computational tools to identify drug or vaccine candidates for treatment is not helpful, since the predictions of structures have not been verified experimentally and need to be validated by clinical trials (5). The main limitation of the use of AI tools is the paradox of many / few data, that is, large amounts of data with low reliability; there is also the right to data privacy (3,6). Knowing the strengths and limitations of AI-based technological tools, they should be developed and applied under rigorous models to combat this pandemic.

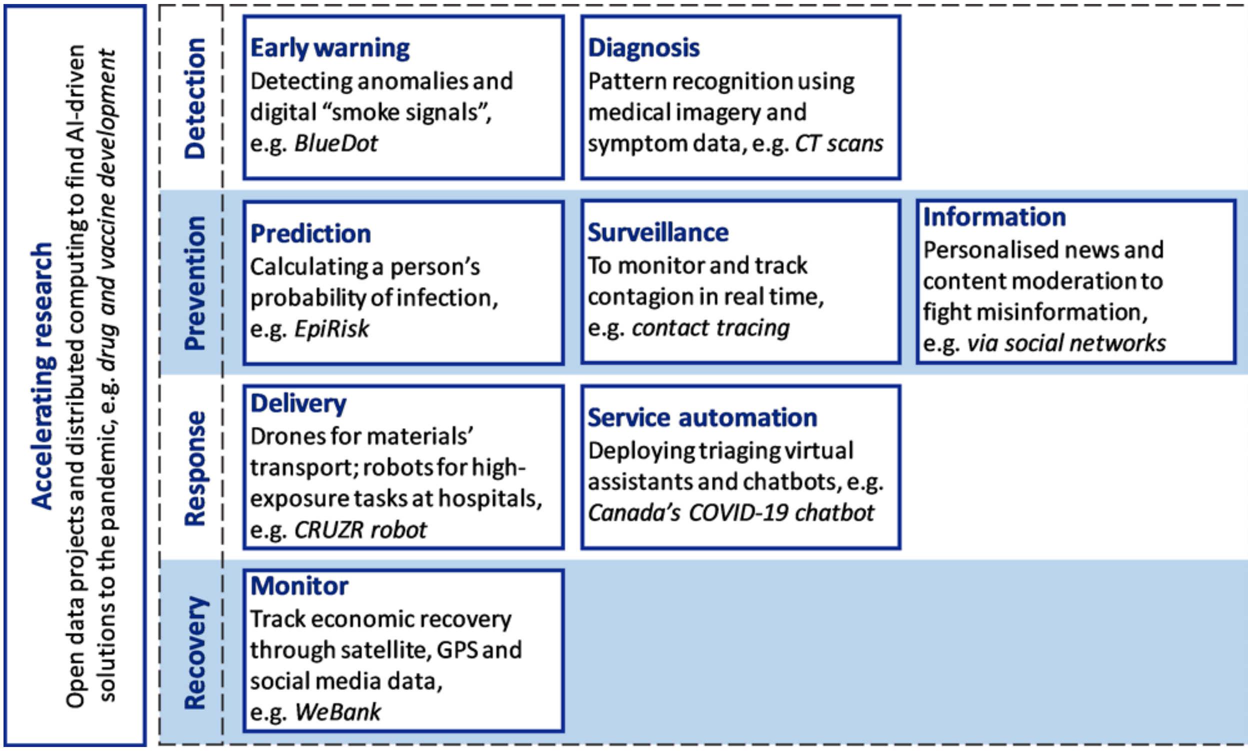



**Figure 3.1:** Covid 19 and AI.

## 3.1 COVID 19 AND MACHINE LEARNING

Currently the world is dealing with a pandemic of um novo coronavirus, severe acute respiratory syndrome coronavirus 2 or SARS-CoV-2, which has become known as COVID-19 (SHOJI et al., 2020), having its first reported case in Wuhan, China, not the end of the year of 2019. Due to its rapid dissemination, it has become a serious problem of public health in the world. In order to diagnose COVID-19, or test that the RTPCR (NISHIOKA, 2020) is being used, but due to its lack in some locations and a delay in obtaining the results, it becomes necessary to identify and develop ferramentas that can help you professionals A number of alternatives to aid in the identification of COVID-19 and the use of chest radiography, which shows characteristics similar to other pneumonia caused by other coronaviruses (HOSSEINY et al., 2020). In the meantime, a rapid radiological interpretation of images is always available, particularly where we have few resources that pneumonia (caused by viruses and bacteria) has a higher incidence and higher mortality rates. The process of interpreting agents (brain tumors or lung anomalies, for example) is a complex activity and, therefore, is necessary, or the use of image processing techniques, often combined with machine learning techniques, to identify applications of deep learning techniques (deep learning) to classify images of x-ray images and a considerable growth in recent years. Several investigations address this topic, such as performing the classification of images to help prevent early diagnosis of tuberculosis, and classifying injuries through chest radiography also used for classifying x-ray images of the x-ray with pneumonia and other injuries.

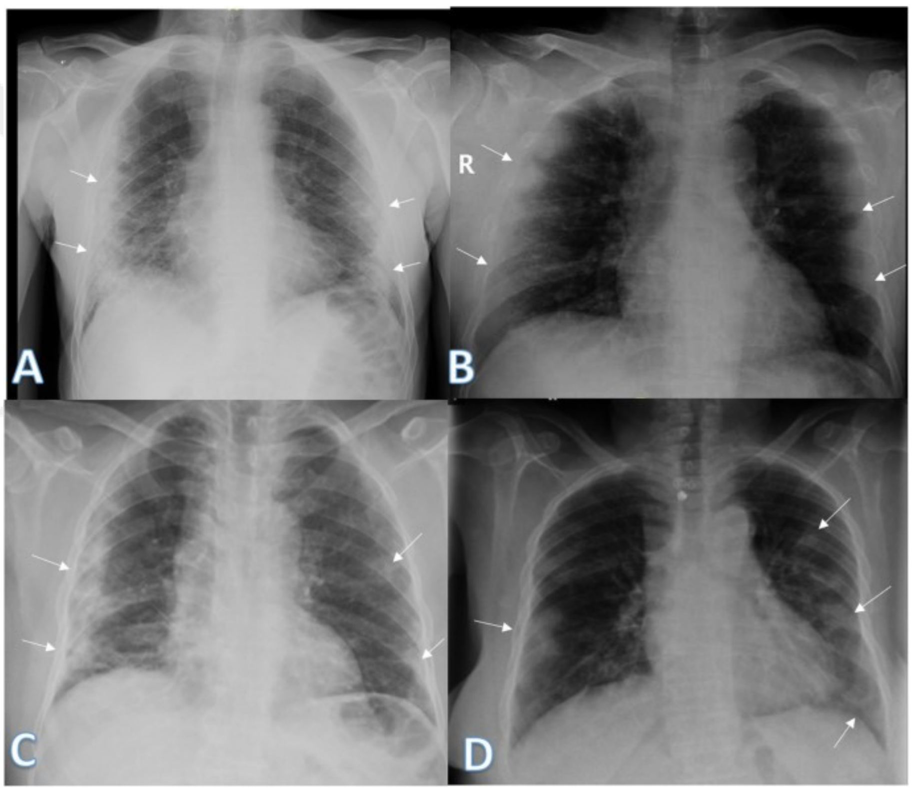



**Figure 3.2:** Effects of COVID19 in X-rays.

Machine learning is defined as an artificial intelligence application, where available information is used, through algorithms, to process or assist the processing of statistical data. However, although the concept of machine learning involves automation concepts, it still requires guidance from a human operator. Thus, machine learning techniques represent an alternative to traditional statistical methods, where most of these techniques are designed to find models that better fit the database available for analysis, but with the exception that They are not restricted to probabilistic models, thus expanding the range of possible solutions to be found, also generating more efficient processes. From the foregoing, it follows that machine learning represents a more flexible way of applying data analysis methods, which is of great importance considering the enormous amount and sophistication of data available today for different systems and teams, giving step into the era of Big Data [2]. There are two main classes of techniques for machine learning: supervised machine learning and unsupervised machine learning. For a case that has a set of data analyzed through algorithms that use machine learning, the data will be classified according to certain desired attributes, thus generating different sets in which its elements share said attributes. The latter can be continuous, categorical or binary. In the event of a case in which the different data sets are classified, the learning of the algorithm will be called supervised, while the unsupervised data does not have a previous classification [7].

## 3.2 DEEP LEARNING

The deep learning paradigm is a sub-area of machine learning that addresses or automated learning, being carried out through the use of successive deep layers within a neural network architecture (FRANCOIS, 2017). As a result, each litter passes or the result of its learning for the next litter. The greater the quantity of litters, the deeper the neural network becomes. A main characteristic of deep learning is the use of litters to perform the extraction of characteristics and classification of two dice (RIGHETTO, 2016). As a result, deep learning algorithms have become viable options for image classification tasks, for example.

Deep Learning, a branch of Artificial Intelligence, is a Machine Learning method that allows computers to solve problems that they would otherwise be unable to solve. Despite the fact that a principled approach for training deep networks has been accessible since the 1980s, it was unable to scale to broad networks, and Neural Networks science entered a time of dormancy. By obtaining a state-of-the-art result on the MNIST handwritten digit dataset in 2006, Hinton GE et al. [1] demonstrated that deep neural networks could potentially be trained effectively. This paper was a game-changer, reigniting and reinforcing faith in Deep Learning. For the first and only Deep Learning-based method, Krizhevsky A et al. [2] won the annual ImageNet Large Scale Visual Recognition Challenge by a large margin in 2012. This was the first time modern Deep Learning was introduced, and it has since been created and implemented in a variety of applications. Deep Learning is now used in a variety of applications, ranging from spam filtering to ad recommendation systems, facial recognition to self-driving vehicles, market price prediction to music generation.

### 3.2.1 Convolutional Neural Network

RNC are biologically inspired architectures capable of being trained and learning / generalizing invariant representations of scale, translation, rotation and related transformations (LIU; ROSENBERG; ROWLEY, 2007; JURASZEK et al., 2014). The RNCs comprise one of the types of algorithms in the area known as deep learning and are designed for use with data of multiple dimensions, making them good candidates for solving problems involving image recognition (AREL et al., 2010). Similar to traditional Computer Vision processes, an RNC is able to apply filters to unstructured data, maintaining the neighborhood relationship between the pixels of the image throughout the processing of the network. Figure 1 shows the structure of an RNC.

RNCs have been used by the scientific community in medical imaging because of their excellent performance demonstrated in Computer Vision. Usually, as shown in Figure 1, three types of layers are used to build an RNC (MOUSSER; OUADFEL, 2019): convolutional layer, pooling layer and fully connected layer.

In general, an artificial neuron is a function that accepts vectors as inputs, conducts nonlinear transformations on them, and then outputs a value. As each neuron is linked to another neuron in the previous layer, a Deep Neural Network is created by piling these neurons together hierarchically. This architecture is ideal for dealing with tabular details but not for dealing with images in computer vision issues. For example, if the network has a 1000x1000x3 input layer, which is just around a one-megapixel picture, and then just one 1000-unit layer, the total amount of parameters is already three billion. This method is not scalable since the computing expense and memory requirements skyrocket as the picture size grows. The Convolutional Neural Network (CNN) is a form of Deep Neural Network that is particularly useful for computer vision tasks. Images are fed into CNNs, and then process them using convolutional operations to provide a final vector that summarizes the image's most interesting attributes. Following that, the vector would be fed into a sequence of Fully Connected Layers for classification. A layer of ten 3x3x3 filters has just 280 parameters, and this amount would remain constant as the input image size grows, allowing for the training of deeper and larger networks. The design of AlexNet, the first CNN to win the ImageNet competition in 2012, as seen in Figure 1. It had error rates of 15.3 percent [2, 7], while the runner-up had just 26.2 percent. AlexNet has eight layers, five convolutional layers, and three totally linked layers, as seen in Figure 1. The height and width of the network decline when you go further into it, although the amount of channels (depth) rises. A typical pattern of CNN architecture is a sequence of convolutional layers, accompanied by pooling layers, and finally a few completely linked layers. AlexNet has used two GPUs and ReLU instead of Sigmoid as an activation feature, making training quicker [2, 3]. CNNs have been extensively used in computer vision problems such as target identification, semantic segmentation, facial recognition, and self-driving vehicles since then.

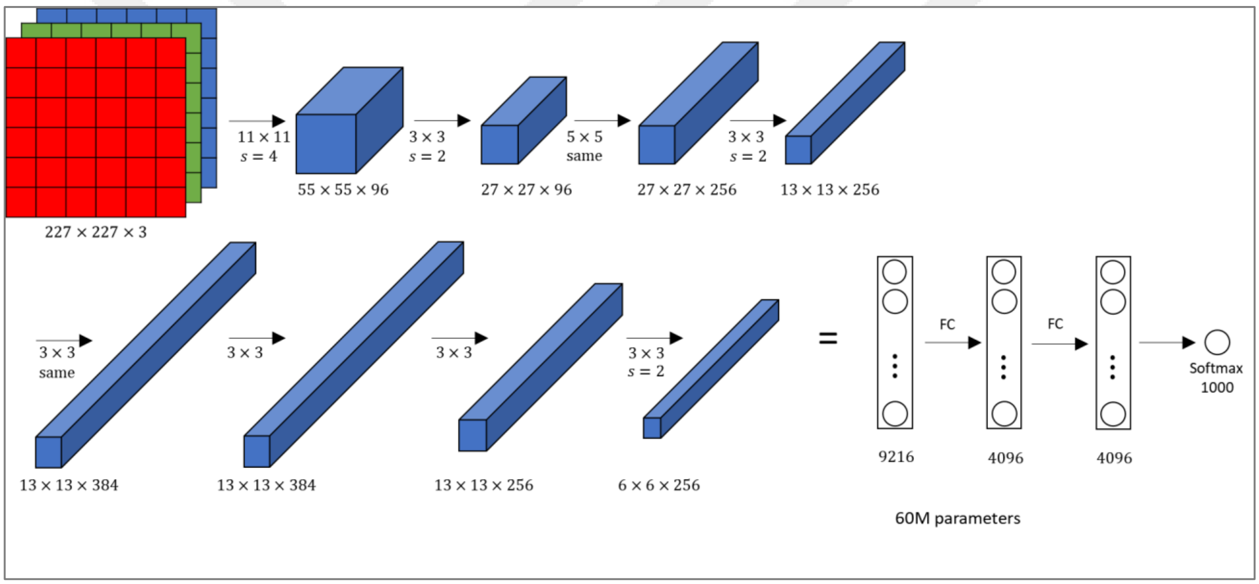



**Figure 3.3:** CNN AlexNet Image Net.

There are four major explanations why deep learning has been so common. First, because of the large number of sensors and the digitalization of culture, it is simpler to gather data as people invest a lot of time online. Standard Machine Learning models are incapable of dealing with greater databases, while Deep Learning methods can continue to improve as more data is collected. (4), (20). Second, advances in computer hardware, especially gpu (GPUs), contributed to a significant reduction in training time. Researchers will deploy and train a network after getting an idea for one, then fine-tune the design given past performance. This method becomes strongly iterative when the preparation period is shortened from hours to minutes. Third, advancements in software also aided the growth of Deep Learning. Training Deep Neural Networks has been quicker thanks to improved algorithmic approaches such as flipping from

Sigmoid to Linear Transfer Units as the activation feature. (2), (3). Deep Learning libraries including Tensorflow and Pytorch also lowered the barrier to entry such that not just academics but even students and clinicians may use it to deal with their problems. Finally, after deep learning's breakthrough in computer vision in 2013, there has been a flurry in industry funding from both startups and software behemoths. Google paid more than $500 million for DeepMind, an AI company, in 2014. Google CEO Sundar Pichai confirmed in the 2015 Alphabet earnings call that machine learning will be applied to all of their brands. (6,22) As a consequence of this flurry of spending, the amount of Deep Learning employees has risen.

### 3.2.2 SVM Classification

SVM is one of ML's classification algorithms that allows you to obtain the expected output for each supplied input. It is mostly applied to the binary and linear type of classification, although it can be extended to multiclass classification. Binary because it allows distinguishing between two classes and linear because the equation that defines the classification is a straight line or a two-dimensional space. This algorithm has been referenced by many classification problems due to its computational efficiency and ability to handle high-dimensional feature spaces, for which it is considered one of the best algorithms available [14]. The particularity of SVM is that the calculation of the solution equation does not take into account all the data in the sample, but only considers a limited number that have specific and singularly important properties, called "Support vectors. The planes divide two sets of red and green color, For example, if we talk about the classification of photographs where it is only necessary to obtain those that present cars, then the red set can represent the images that contain at least one car and the green set those that do not contain any. Research has focused on building Machine Learning models by fine-tuning the training characteristics in existing databases (collected in past years) and others have focused on collecting new data. All this with the aim of finding the appropriate model that minimizes or cancel the false positives in the prediction. Among the proposals, is the application My Page Keeper [18] par to Facebook, which once installed by the user, consumed their input and output data. This data fed the SVM algorithm and its aim was to alert users to socware5 content whenever a post was received on the profile before it appeared publicly. The application was installed by 12 thousand users whose publications were observed in a period of four months. Another investigation was focused on the detection of malicious accounts [19]; A tool called

COMPA was developed that analyzed changes in user behavior, for example: excessive publications, divergent topics and number of friend requests sent.

Support Vector Machines (SVM), is a type of supervised machine learning based on statistical learning theory, which can be used in pattern recognition and regressions or in data classification. It allows to build highly complex models, but which at the same time represents simplicity when it is analyzed mathematically, since it behaves like linear algorithms in a multidimensional space [8]. The initial approach to a problem to be solved through SVM begins by having an initial set of training data of the form:

$$D = \{(\mathbf{x}_\mathrm{i}, y_\mathrm{i}) \in X \times Y\}, \quad \mathrm{i} = 1, l$$

Where l corresponds to the number of data pairs available, x is a high-dimensional input vector and y corresponds to a scalar associated with vector x. The objective then is to find a mapping function y = f (x), which allows relating the training data of the set D, in order to be able to predict or calculate the value ye = f (xe) from any vector xe and that it does not necessarily belong to D. [9]. Thus, SVM can be used to perform regressions that allow obtaining the function f (x) that fits the training data D, or also for classification from a label (for example y ∈ {1, −1}), where you want to know the sign (positive or negative) of the function obtained from an input vector x.

For any case study, the way in which the data can be classified is directly related to the nature of the data. Thus, it is possible to have data separable by a straight line, a parabola, or more complex functions in the case in which we have a set of vectors x∈ Rn, associable to any of two classes and ∈ {+1, −1}, we want to find a function f (x) that allows classifying new data according to the value it returns the evaluation of the vector x in this, sign (f (x)). An example is illustrated in Figure 3.4 where each vector x = (x1, x2) is associated with one of two existing classes, circle or square (+1 and −1). It is then sought, from an initial data set (for example, those shown in Figure 3.4), to find a linear hyperplane that has the shape shown in Equation 3.2. that separates the two classes in a better way, maximizing the distance to the points closest to it.

$$w^T x + b = 0$$

Those vectors closest to the margin of the hyperplane are called Support Vectors (SV) and they define the values that both the normal of the w plane and the displacement of the hyperplane with respect to the b origin should take.

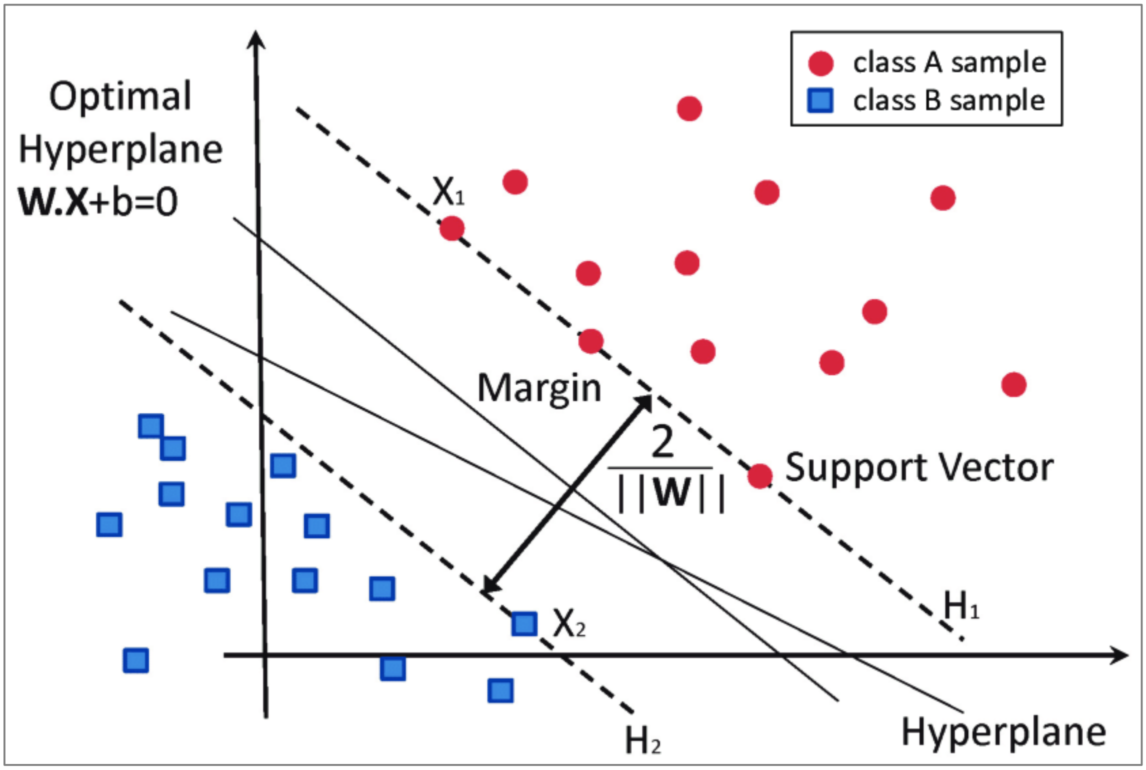



**Figure 3.4:** SVM hyperplane.

Once the machine is trained, a decision function is obtained in the form shown in Equation 3.3, which for any input vector (x1, x2) gives a scalar value, which will classify the vector according to the sign that yields the function d.

$$\mathrm{d}(x, w, b) = w^T x + b = \sum_{\mathrm{i}=1}^{n} w_{\mathrm{i}} x_{\mathrm{i}} + b$$

Now, given that the possible values of w are infinite, we must find the one that fulfills the restriction of maximizing the margin M between both classes, given by 2 || w ||, which is equivalent to solving Equation 3.4, corresponding to the distance that separates the support vectors of different classes.

$$\min \quad \frac{1}{2} w^T w$$

In other words, for each SV it is satisfied that yi [w T xi + b] = 1. so for each vector xi (3.5)) is satisfied, which represents the restrictions to be able to solve the quadratic problem represented by (3.4).

We can then write the LaGrange an of the problem, introducing a Lagrange multiplier αi associated with each training vector xi for the machine, thus leaving Equation 3.6, where by obtaining the minimum with respect to aw and b, and the maximum with respect to αi not negative (αi ≥ 0), an optimal saddle point is reached.

$$L(w, b, \alpha) = \frac{1}{2} w^T w - \sum_{\mathrm{i}=1}^{l} \alpha_{\mathrm{i}} \{ y_{\mathrm{i}} [w^T x_{\mathrm{i}} + b] - 1 \}$$

In order to facilitate the solution, the Karush-Khun-Tucker (KKT) conditions can be used to establish the dual of the system [10], which are sufficient and necessary for the maximization of 3.6, given that both this and the constraints are convex. Thus, applying the derivatives for minimization or maximization of 3.6, as appropriate, it is obtained that:

$$\frac{\partial L}{\partial w_0} = 0 \implies w_o = \sum_{\mathrm{i}=1}^{l} \alpha_{\mathrm{i}} y_{\mathrm{i}} x_{\mathrm{i}}$$

$$\frac{\partial L}{\partial b_0} = 0 \implies \sum_{\mathrm{i}=1}^{l} \alpha_{\mathrm{i}} y_{\mathrm{i}} = 0$$

Replacing the above in the primal LaGrange an, we pass to the dual function, thus leaving the problem of maximizing equation 3.10 with respect to αi, subject to the restrictions set out in Equation 3.11. It is possible to notice immediately that the dual depends clearly on the scalar product of the training data, as well as on the multipliers that accompany them.

$$L_{\mathrm{d}}(\alpha) = \sum_{\mathrm{i}=1}^{l} \alpha_{\mathrm{i}} - \frac{1}{2} \sum_{j,k=1}^{l} y_j y_k \alpha_j \alpha_k x_j^T x_k$$

Where NSV represents the number of SVs found in the optimization, which in turn are those points of the training data that have a Lagrange multiplier other than zero, in other words, both w0 and b0 only depend on the SV's found. Thus, the decision hyperplane D (x) (3.14) is obtained, whose sign found for a given input vector x will indicate to which class it belongs.

$$D(\mathbf{x}) = \sum_{\mathrm{i}=1}^{l} w_0 x_{\mathrm{i}} + b_0 = \sum_{\mathrm{i}=1}^{l} y_{\mathrm{i}} \alpha_{\mathrm{i}} \mathbf{x}^T \mathbf{x} + b_0$$

The case presented above can also be applied to problems in which the data classes present a small overlap 9, through slack variables that allow an error in the classification margin. An example of this can be seen in Figure 3.5, where it is illustrated that given any linear separation plane, there will always be vectors (x1, x2) that remain within the margins established by the support vectors (that is, they are misclassified), and that in this case correspond to the squares and circles between the dotted lines and the solid line.

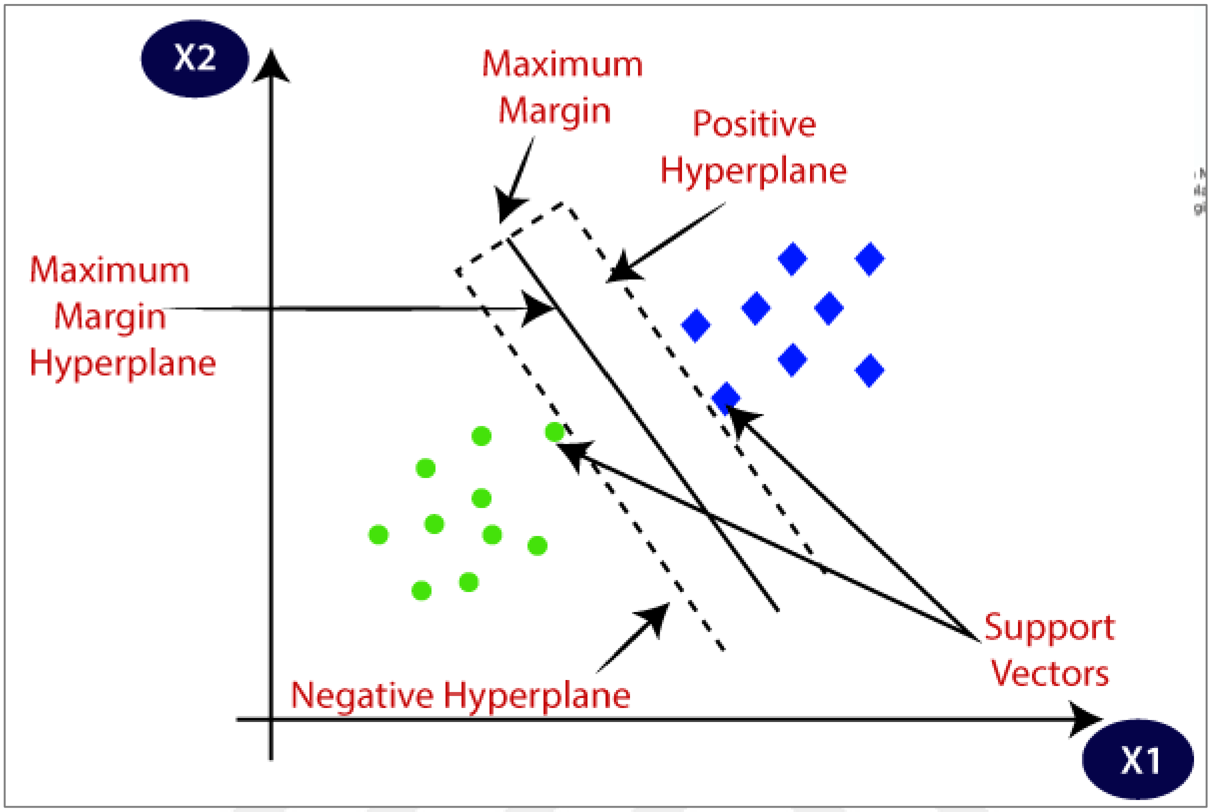



**Figure 3.5:** SVM classification using hyper parameters.

However, the amount of misclassified vectors represents a small portion compared to the trend. In this case, it is not possible to use the linear separation procedure directly, since those variables that are not outside the range determined by the SVs will have Lagrange multipliers αi that will tend to infinity, thus indeterminate the solution of the problem. Thus, by introducing slack variables ξ, a misclassification of the data is allowed in exchange for a penalty for those vectors that are on the 'wrong' side of the decision edge, controlling the smooth linear margin by a parameter C that determines the cost or 'trade-off' of having points with incorrect classification, so the problem raised in Equation 3.4 must be updated to now have the lowest possible cost in data classification, that is, Equation 3.15 is present.

$$\min \quad \frac{1}{2}w^T w + C\sum_{i=1}^{l}\xi_i$$

$$y_i[w^T\mathbf{x}_i + b] \geq 1 - \xi_i, \quad i = 1,..l; \quad \xi_i \geq 0$$

That is, a Lagrange multiplier βi must be added for the new restrictions that govern ξ, where, as in the previous case, the problem is based on finding the saddle point of the LaGrange an raised in Equation 3.17.

$$L_p(w,b,\xi,\alpha,\beta) = \frac{1}{2}w^T w + C\left(\sum_{\mathrm{i}=1}^{l}\xi_{\mathrm{i}}\right) - \sum_{\mathrm{i}=1}^{l}\alpha_{\mathrm{i}}\{y_{\mathrm{i}}[w^T x_{\mathrm{i}} + b] - 1 + \xi_{\mathrm{i}}\} - \sum_{\mathrm{i}=1}^{l}\beta_{\mathrm{i}}\xi_{\mathrm{i}}$$

Performing the derivatives to maximize and minimize as appropriate, the dual shown in Equation 3.18 is proposed, which again only depends on the training data, as well as on the multipliers associated with these, leaving the restrictions of the slack variables alone in proposing the possible values for αi.

$$L_{\mathrm{d}}(\alpha) = \sum_{\mathrm{i}=1}^{l}\alpha_{\mathrm{i}} - \frac{1}{2}\sum_{\mathrm{i},j=1}^{l} y_{\mathrm{i}}y_j\alpha_{\mathrm{i}}\alpha_j x_{\mathrm{i}}^T x_j$$

In classification problems that arise in reality, many times there is not only the presence of overlap, but it is definitely not possible to separate the groups or classes through a hyperplane. Such is the case of the example illustrated in Figure 3.6, where it is observed that any attempt to classify from a straight line will generate a significant amount of misclassified data, thus making the classification error for new data very high, and therefore not very useful for what is desired. Fortunately, the previously introduced methods have great flexibility in adapting to this type of data and creating non-linear decision edges. Given that the data xi are presented as an internal product in the solutions shown above, it is possible to obtain an alternative representation of the data by mapping them to a space of different dimensionality called the feature space 10 [11], performing the replacement shown in Equation 3.21, where Φ (·) is the mapping function such that the data can be separated in a linear way in the new space of sufficiently high dimensionality.

$$\mathrm{x}_{\mathrm{i}} \cdot \mathrm{x}_j \rightarrow \Phi(\mathrm{x}_{\mathrm{i}}) \cdot \Phi(\mathrm{x}_j)$$

### 3.3 IMAGE SEGMENTATION

In computer vision applications such as video surveillance or human movement analysis, the ability to extract objects of interest from a video sequence is a crucial preliminary task. Video surveillance is a key technology for public safety (for example, in transport networks, city centers, schools and hospitals), efficient management of transport networks and public services (for example, traffic lights, road crossings), recognition of patterns of human behavior, etc (Maddalena and Petrosino, 2008). Frames are typically captured from a scene using a static camera that compresses the video information. In this scenario, detecting the incoming objects is an essential stage in the analysis of the scene, since a large number of characteristics can be extracted from the foreground to perform classification and recognition tasks in subsequent phases. A common assumption is that the frames of a scene without objects of interest show a regular behavior, which can be described with a statistical background model. A typical approach to discriminate objects of interest from the background is detection by a background subtraction process. The idea of this process is to remove the current image from the reference background model. A pixel in the current image is considered a background when its observed value (color) is well represented by the model. In another case, the pixel is considered the foreground. However, the background pattern is a problem in applications with difficult circumstances, such as lighting changes, moving trees, water, video displays, rotating fans, shadows, camouflage (foreground objects are similar to the background), changes occasional from the true background (for example removing an object from the scene), traffic, etc.

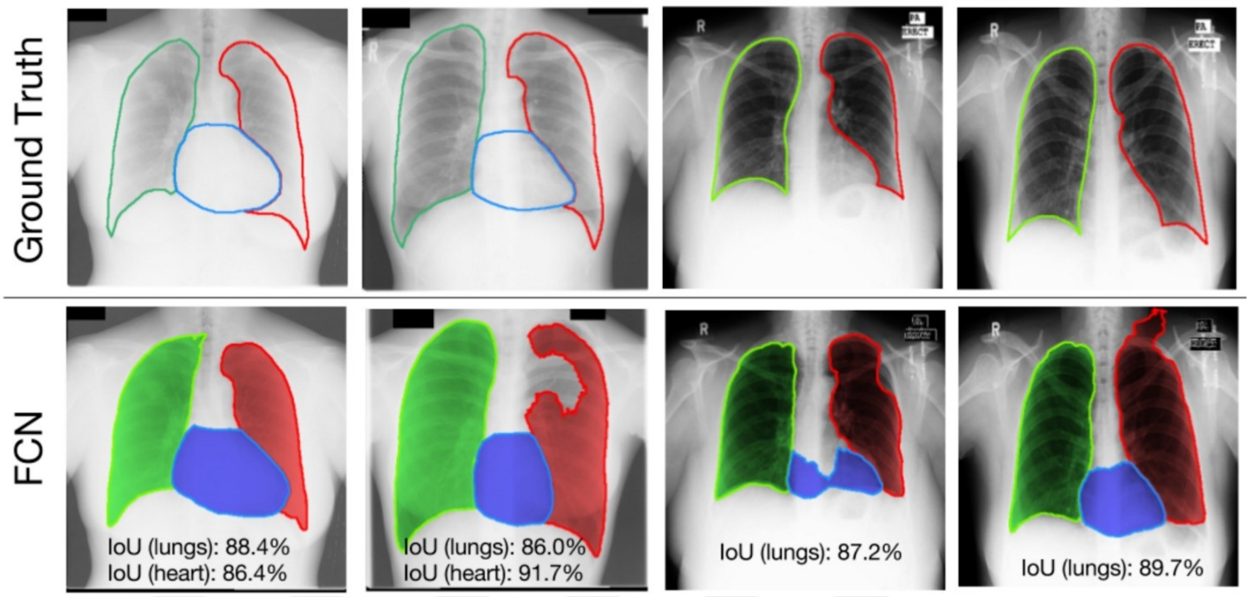



**Figure 3.6:** Lung X-rays segmentation.

These problems cannot be solved by simple static background models. There are many solutions that handle the dynamic background problems mentioned above. But the computational cost of many of them is expensive, which hampers their use in real-time applications. Even the case when a visual analysis is used to further evaluate the detected events, the accumulated data can become too much for a posteriori process. Therefore, many systems require real-time processing to maintain the ritual of the incoming video data stream. Neural networks have been developed that offer solutions in the field of background modeling (Maddalena and Petrosino, 2008; López-Rubio et al., 2011a). In this chapter, we propose a neural model that is a special case of self-organizing maps. It is specially designed to handle unstable input distributions with a two-phase learning strategy, where the winning neuron executes a more intense It is an adaptation to the input pattern than the remaining neurons, which learn slowly to follow changes in the input distribution. The algorithm of the proposed learning model is designed to optimize a cost function that considers these two objectives explicitly. Therefore, the original contribution of the proposal is twofold: An unsupervised learning neural network model is proposed, which is specifically designed to manage the needs of variable input distributions. The proposed learning model is applied to the problem of background modeling, which is one of the key problems in the development of current video surveillance systems.

In particular, the problem of neurons ceasing to represent some input, that is, the problem of dead neurons, is not solved by a standard self-organizing map topology. Rather, a new term is added to the cost function to move all neurons slightly to the center of the input distribution. Therefore, all neurons follow the input distribution as it varies over time, while a non-fixed topology is imposed on the neural network. This also differs from other ART-type background models in two ways. For one thing, the new proposal does not need a surveillance parameter to avoid dead neurons. On the other hand, in the new proposal all neurons are used to model the background of the scene, which causes a better use of them and eliminates the need for a procedure to decide which neurons are associated with the background.

In this section we propose a neural network that is a special case of self-organizing maps. It is suitable for learning non-stationary input distributions. The greatest difficulty in adapting a competitive neural network to this type of problem is that, even if a neuron has correctly positioned its prototype at the center of a group, the group could disappear as time passes. Therefore, it is possible that the neuron stops representing any current input sample, which would lead to its death, that is, its prototype would not win the competition anymore, so it would not learn. To get around this problem, it is proposed that all neuron prototypes move a little towards the current center of the general input distribution. In this way all the prototypes stay close to the current samples in the entrance space, which offers them a better opportunity to represent a group. This strategy is implemented by adding a term to the cost function of the means standard, so there is a small penalty for the squared Euclidean distance from each prototype to the current input sample. The reason is that the term extra ensures that even if a neuron does not win the competition for a long time, it will move towards the center of the input distribution where there is a better chance of finding samples to represent. Let D be the dimension of the input space, that is, the input samples are x ∈ R D. The RGB color space has been adopted, therefore, in this case D = 3. As is done in traditional competitive learning, N neurons with N> 1 are considered, where each neuron i stores a prototype vector wi ∈ RD, with i ∈ {1, ..., N}. The following cost function is considered: [11].

$$\mathcal{E} = \frac{1}{2}\sum_{j=1}^{M}\left\|\mathbf{x}_j - \mathbf{w}_{Winner(\mathbf{x}_j)}\right\|^2 +$$

$$\frac{\lambda}{2}\sum_{j=1}^{M}\sum_{i=1}^{N}\left\|\mathbf{x}_j - \mathbf{w}_i\right\|^2$$

It is well known that standard competitive learning is guaranteed to reach a local minimum of the Mean Square Error (Mean Squared Error which is the first term in equation (3.1). This alerts strategies to escape these minimal premises. The second term proposed in equation (3.1) fulfills this role, since it drags all the prototypes a little from their respective group centers, which are associated with the local minimum of the Mean Square Error, to the center of the input distributions. This dual learning scheme based on two cost terms with different objectives facilitates the search for alternative group configurations. Competitive learning neural networks can be used for the partitioning of groups that is, each sample belongs exactly to one group. In this case, the number of neurons is fixed to achieve real-time functioning, contrary to grouping methods that allow the division or grouping of groups (Chira et al., 2014). The group of input samples associated with neuron i is defined as follows:

$$C_i = \left\{ \mathbf{x}_j \in \mathbb{R}^D \mid i = Winner\left(\mathbf{x}_j\right) \right\}$$

# 4. PROPOSED METHOD

More than one million adults are hospitalized with pneumonia, and about 50,000 die from this disease in the US alone each year [6]. Chest radiographs or films are currently the best method of diagnosing pneumonia [7], and they play a crucial role in clinical care and epidemiological studies. However, detecting pneumonia on radiographs is a difficult task that depends on the availability of radiologist specialists, so there is a wide opportunity to help specialists with new Artificial Intelligence technologies to facilitate their work and improve the health system. As mentioned above, chest film is used in the screening and diagnosis of many diseases of the lungs. A large number of radiograph studies and radiological reports are typically stored in the PACS (Image Archiving and Communication System) systems of hospitals. A good deal of research has been done to take advantage of the knowledge contained in these databases, but there is a very specific challenge: medical records are often loosely labeled and do not contain annotations. This means that the x-ray is associated with the diagnosis of the specialist doctor -as negative (healthy patient) or positive (identifying the disease). But the radiograph does not contain delimited the region of the image that led him to such a conclusion or diagnosis. So recent research has focused on exploiting these data sets with deep learning paradigms, which require large amounts of data or images, building large-scale CAD systems for medical purposes. Detecting chest diseases and especially pneumonia on x-rays can be difficult for the specialist. The appearance of various plaque diseases is often vague, can overlap with other diagnoses, and can mimic other benign abnormalities or other diseases. Some of the most common diseases and the approach of this and other state-of-the-art approaches are shown in Figure 4.1. As can be seen, the quality of many of these images is suboptimal and not suitable for machine learning algorithms.

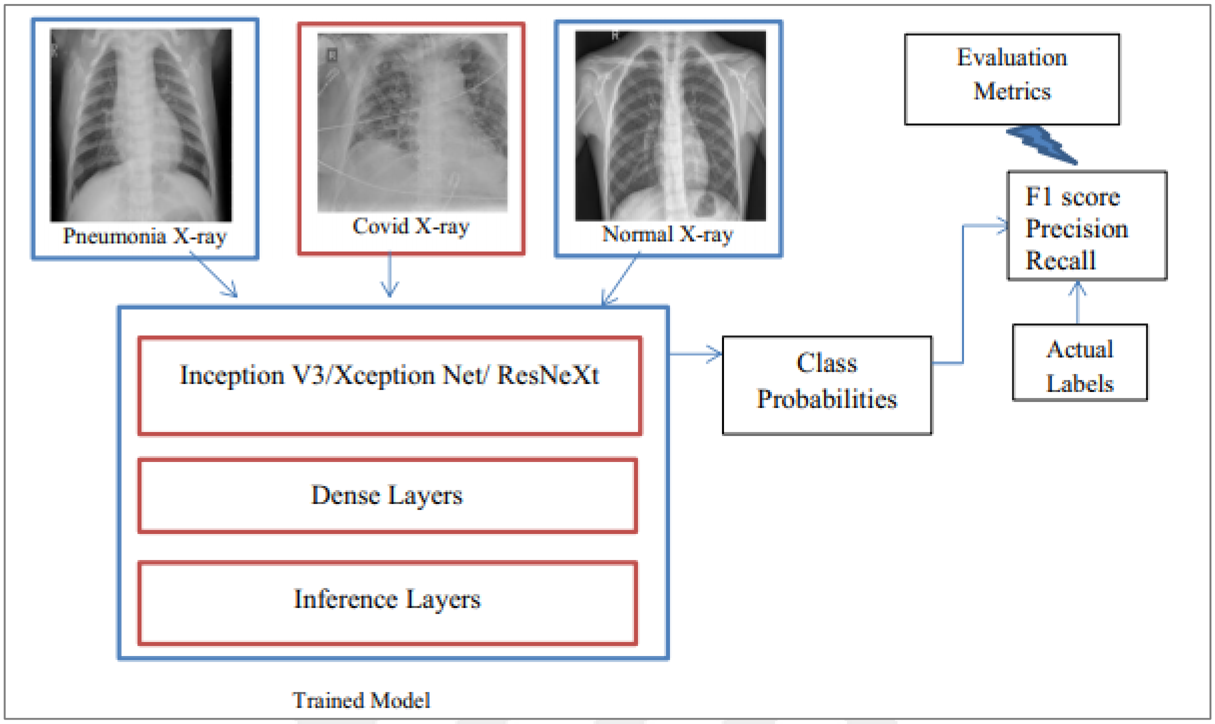



**Figure 4.1:** Training model for the Covid-19 detector.

Furthermore, generic and open anatomical and pathological labeling cannot be obtained through crowdsourcing or public outsourcing, since it is implausible for non-medically trained annotators. Also, complete annotation at the region level in bounding boxes to locate the specific pathology, which would be necessary in data sets for computer vision [8] is currently not feasible. Consequently, most of the research in the identification of chest diseases is formulated as a weakly supervised multi-label classification problem, to overcome the aforementioned difficulties and using the information from the complete image in tandem with weak image annotations and scanning different avenues of investigation.

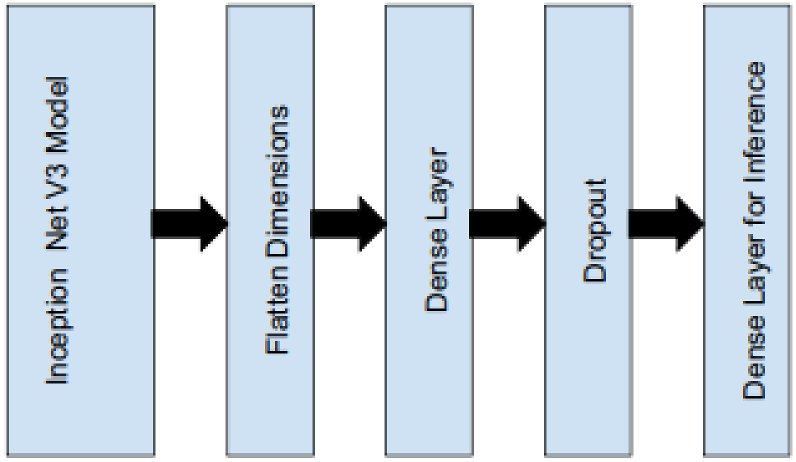



**Figure 4.2:** Data Entry Form Covid-19.

In order to alleviate the paucity of publicly available data sets, there have been several recent efforts to create them [9], with patient numbers ranging from a few hundred to up to two thousand. Particularly for chest film images, the largest public dataset was Open-I [10], which contained 3,955 radiology reports from the Indiana Network for Patient Care and 7,470 linked chest films from hospital PACS systems. More recently, Wang et al. (2017) presented a new very large database of radiographs, “ChestX-ray8” [11]. This database is composed of 108,948 frontal view X-ray images of 32,717 unique patients with labels of 8 diseases obtained using text mining on the radiological report and where each image could have multiple diseases. This dataset has since been used in several studies using loosely supervised approaches to deep learning model training. Due to careful healing and large volume, ChestX-ray8 represents a more realistic and comprehensive clinical study than Open-I, and consequently more suitable for model development and comparison. Furthermore, it has become the de facto standard upon which other methods have trained and validated their models. Several papers have attempted to solve the problem of image annotation on chest films [12], including the initial study presented together with the data set. Most of these studies, including the one presented in [13], make use of ConvNet RNN models in order to model dependencies between labels. which are very strong when evaluating chest diseases Additionally, most of these studies detect pneumonia and locate the areas in the image most indicative of this pathology, creating a heat map in the process, as illustrated in Figure 2 above. Compared to previous work, our approach differs in three main ways: 1. We make use of data increase to expand the size of the Chest X-ray 8 dataset. 2. We implemented a data filtering system to reduce images that could negatively affect the training process, and 3. We designed a much smaller and decreased convolutional network that achieves better results than those given by Wang et al [2017]. In the next section, we will explain each of these contributions in detail, and in sections 3 and 4, we will thoroughly discuss the results obtained by our methodology. Finally, we will discuss the limitations of our approach and future lines of research...

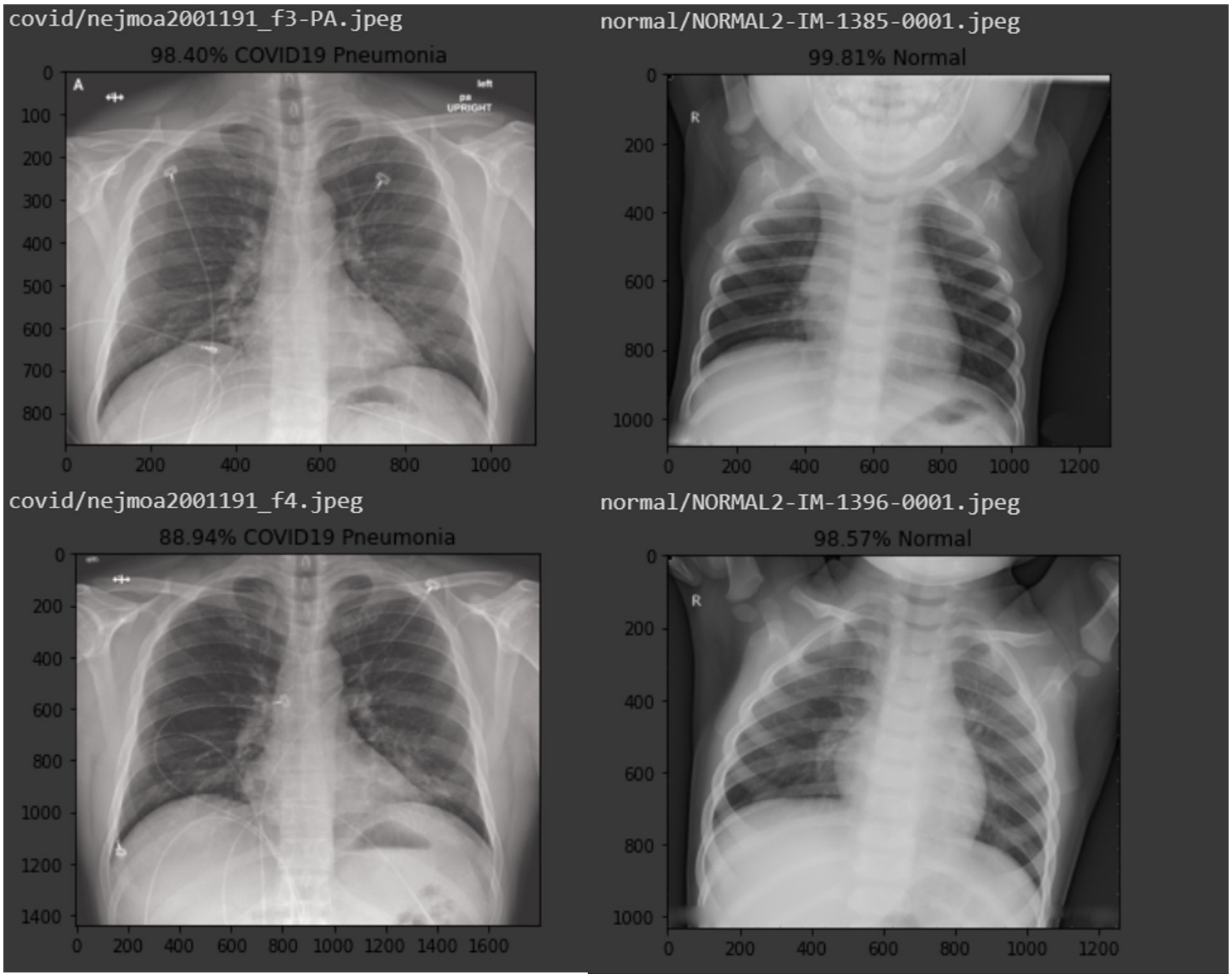



**Figure 4.3:** X-ray Model After Modification.

The approach presented here in this paper is multiple classification problems, where we identify and classify the X-ray image into 8 potential classes or diseases. Then we take the data from the Chest X-ray 8 array and train it through a convolutional network to do the classification process, getting a model after some subtle modifications such as dropping out or getting rid of weights and organizing the L2, among other strategies. However, deep learning is not without problems. One of the biggest problems with deep learning when working with data sets like the one used in this study is the problem of overtraining. Deep learning needs a lot of training. One widely used solution is data augmentation (A.D.), giving and generating more and more training data from existing samples. and work on applying a variety of random transformations that produce reliable images to these examples. In deep learning training, data augmentation can greatly reduce validation loss and give correct results, and this is shown as shown in Figure 4.3, where the graph is shown with the training and validation data for two identical convolutional networks, one of which was used without data augmentation. The other is using a data augmentation strategy. As clear that the use of data increment reduces and solves the problem of overtraining which can be observed on the left side of the figure, it is also observed that the value of the loss function with the validation data gradually decreases in the beginning and then starts to increase again. While it continues to decrease with training data. The right side of the figure shows how validation loss can be regulated using data augmentation, among other strategies...

Two dual-head cameras are used and The per- and inter-critical examinations are both acquired on the same camera (one or the other, indifferently).

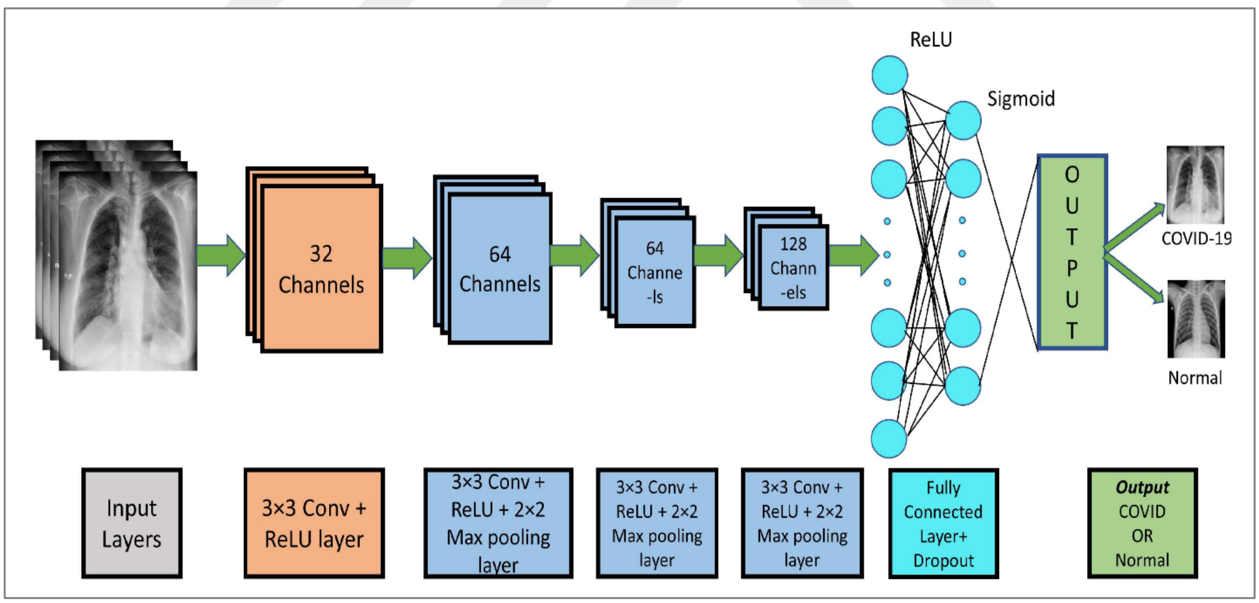



**Figure 4.4:** Infected X-ray Model Covid-19.

However, as we will see in the next section if data augmentation strategies are implemented without regard for the quality of the pictures, some examples of poor training can guide us to a suboptimal form of the convolutional network. The data augmentation approach has proven to be a successful method in different computer vision applications, but as we will see in the next section, there are some important considerations when applying this method to medical imaging. The problem arises from the poor quality of some of the images in the data set, which can affect training and produce a suboptimal model. These models can introduce large numbers of false positives or worse, misclassify a new test sample, as the authors comprehensively document in [14]. The authors of this study applied a battery of tests in which they evaluated the performance of some well-known convolutional networks such as VGG16 and GoogLeNet using images modified with different types of image quality distortions: compression, low contrast, blur and Gaussian noise. The authors found that networks are resilient to compression distortions but are greatly affected by the other three. This has motivated a great deal of research in this area, with some work concentrating on the effects that these distortions produce on image quality for different computational vision tasks based on deep learning networks [15]. This is especially important in the domain of medical imaging, particularly with the ChestX-ray8 dataset, which contains 108,948 frontal radiographs of 32,717 unique patients collected between 1992 and 2015, with labels of 8 common diseases obtained by text mining of radiological reports using natural language processing techniques. As far as we know, no evaluation of the quality of the images has been reported in previous works with this data set, which includes a large number of images with poor quality, as observed in Figure 4, probably introduced by the the way the authors created the data set. Some of the images have very low contrast, while others are out of focus or lack definition, and others suffer from extreme saturation. In addition to the data augmentation strategy described above, the second contribution of this article is the implementation of an image quality assessment to determine if the training example can be considered a good candidate for the data augmentation phase and beyond. processing in the training phase. Some of the considerations for evaluating image quality, as well as the algorithms used to determine it.

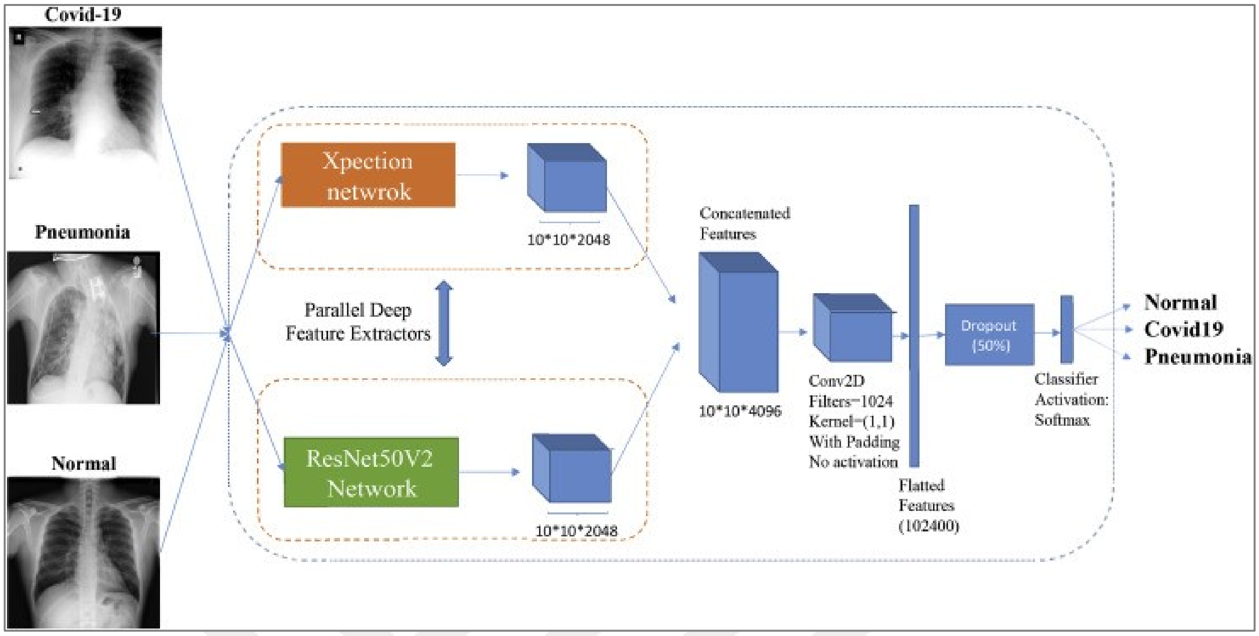



**Figure 4.5:** X-ray Model Not Infected with Covid-19.

As mentioned previously, the performance of convolutional network models can be severely affected by the quality of the input images; the authors of [14] performed individual tests for Gaussian noise, blur, compression, and low contrast. Individually, even for moderate amounts of noise or blur, the accuracy of the network decreases significantly, and one can only assume that the combination of several of these distortions in the images can produce poorer results. Thus, in this article we implement a “filter” or selection process based on various traditional image processing metrics1 to estimate whether an image can be considered for post-processing or not, such as the signal-to-noise ratio (SNR), the blur index, and the variance of the image to determine the contrast. Typically, these parameters are correlated: an image with high noise levels has low contrast and high blur.

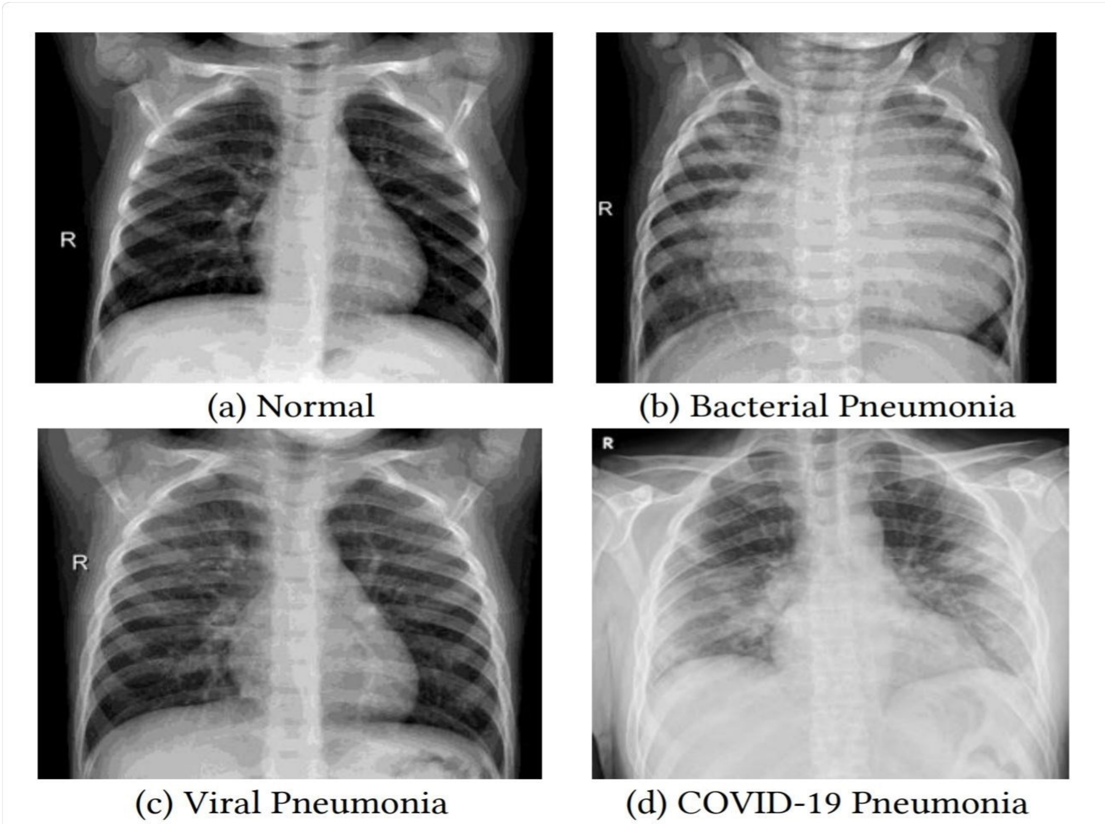



**Figure 4.6:** Types of Viral Infections.

The images in the ChestX-ray8 data were first pre-processed using traditional image processing techniques:

1. We calculated the blur index using a spectral blur estimation operator.

2. To estimate the contrast we use the variance as a measure of image quality, using the Michelson contrast ratio1, as shown in figure 4.5, leaving out images with high variance.

3. Additionally, we performed an SNR test using the methods proposed by Lagendijk and Biemond, to determine if the image is suitable for the data augmentation and training phases. Once these tests have been carried out, we perform the restoration procedures of the image by removing distortions such as blur (using Wiener's convolution), and low contrast (using CLAHE, Contrast Limited Adaptive Histogram Equalization) and repeating the SNR test to determine if the corrected image can be used in post-processing. As we will see in the next section, this process produced an augmented data set where image quality metrics were taken into account.

# 5. SIMULATION AND RESULT

In this section, we will briefly discuss the results obtained with data augmentation via image quality evaluation. As mentioned above, ChestX-ray8 contains 108,948 frontal radiographs of 32,717 unique patients. Due to space considerations, we do not include the full table here, and we focus only on the pathologies that we have decided to address in this article. For a more complete review of the data set, the reader is directed to [9, 11] but we present a brief introduction here. First, the size of the data set was increased to 136,526 images, increasing the number of training examples for some of the previously poorly represented classes (eg, cardiomegaly and pneumonia, to name two cases). Furthermore, most of the examples correspond to healthy patients (63,832) and the rest were divided into 8 classes (as shown in Figure 5.1). The number of images for each of these classes is shown in the first row of figure 5.1 (named “original”) and in the second row we show the number of images that were obtained after the filtering process described and the data augmentation (named "modified").

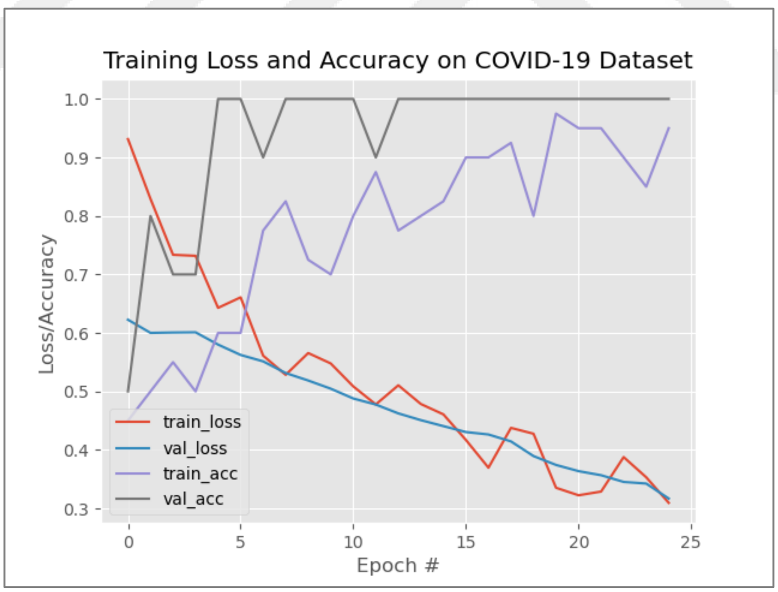



**Figure 5.1:** Training Loss and Accuracy on COVID-19 Dataset.

After applying the data augmentation, we noticed that we got 221.760 images, of which 70,908 were neglected after editing and filtering that evaluated the image quality. Therefore, our modified dataset has grown to a size of 150,852 images. Compared to other datasets in the taxonomy such as COCO and ImageNet, the spatial extent of diseases in Chest X-ray 8 occupies a rather small space of the image, and this may affect the performance of the convolutional network especially when we have limited computing power, like here. The authors of [11] extracted X-ray images from a DICOM file and adjusted the image settings to a size of 1024 x 1024 pixels (from its original size of 3000 x 2000) without losing important details in the image. In our case, the images in the adjusted data set were reduced to a size of 224 x 224 pixels, due to the limitations of the available hardware and their poor quality in dealing with large image sizes, however, we were able to obtain satisfactory results, as we will see below. In the following sections, we will see that we had to make compromises regarding the depth of the convolutional network, to accommodate the hardware used.

**Table 5.1:** Result.

| Label | Precision | Recall | f1-score |
|---|---|---|---|
| Normal | 0.98 | 0.93 | 0.95 |
| Covid-19 | 0.99 | 0.92 | 0.95 |
| Pneumonia | 0.97 | 0.99 | 0.98 |
| Accuracy | | | 0.97 |

However, we will see that we were able to obtain and in some cases exceed the results reported by Wang et al. [eleven]. The convolutional network implemented in our work can be seen in the upper part of figure 5.2. It is a rather minimal architecture, since it is made up of 5 convolutional layers and 3 completely connected layers, with a low parameter count of only 254,138. The architecture is based on Alex Net which has a total of 62.4 million parameters and was gradually simplified until the model was adjusted to the available hardware, an NVidia GeForce 1050 GTX GPU.

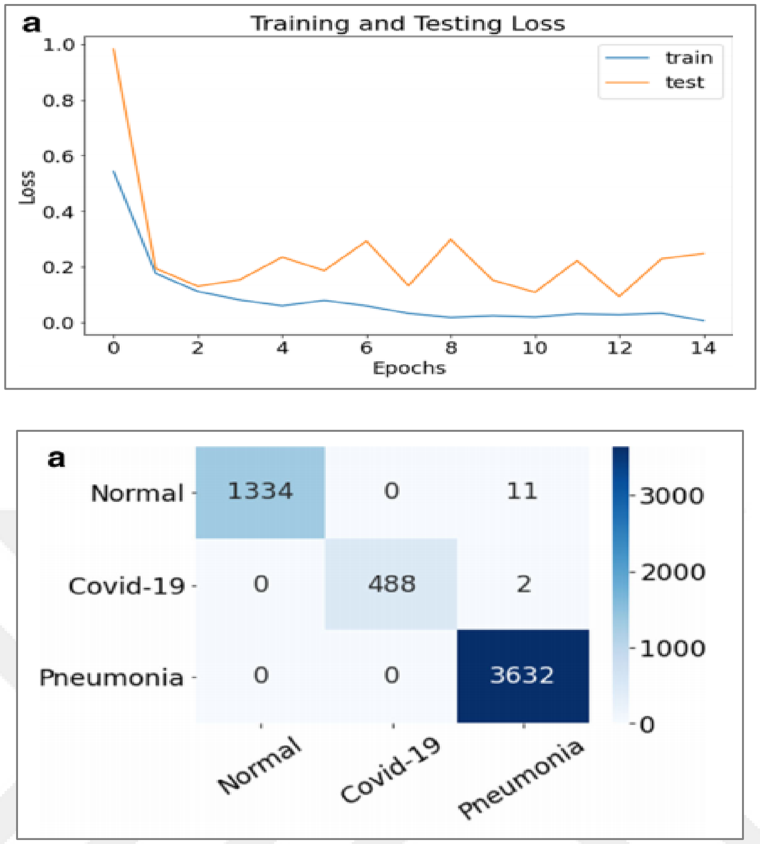
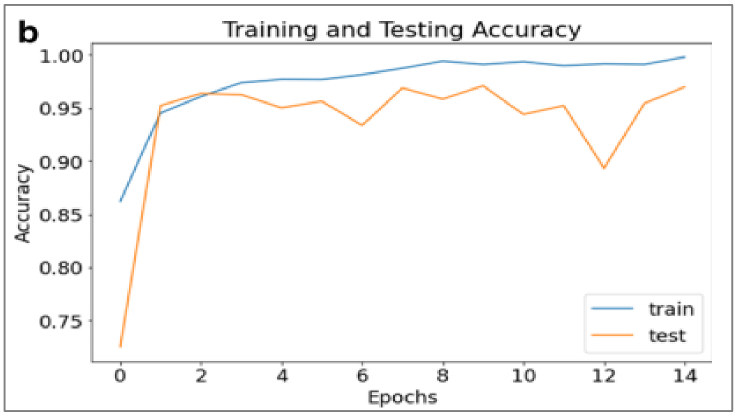



**Figure 5.2:** Training and Testing Loss.

Compare with the architecture used in [11] which tested a set of models and got the best results with ResNet-50, as shown in Figure 6, which contains 25.6 million parameters. Because the image size is large and the GPU memory is poor, it was very important to work on reducing the image size so that the entire network was loaded into the GPU and we increased the number of iterations to collect gradients.



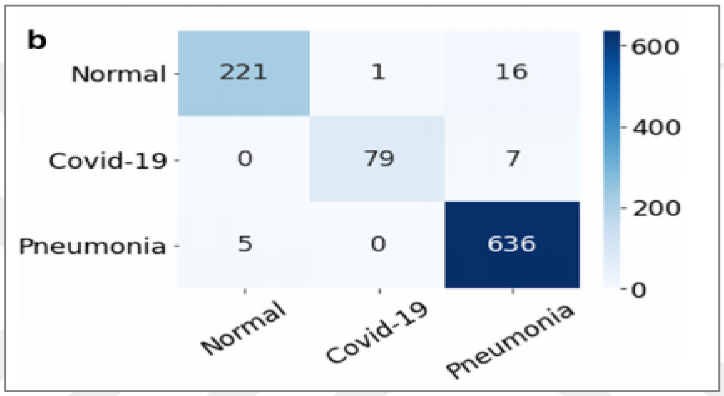




**Figure 5.3:** Training and Testing Accuracy.

Combining the two in different convolutional networks can vary, but in our case, we kept the batch size constant at 32. The network was trained end-to-end using Adam's optimizer with standard parameters (1 = 0.9 and 2 = 0.999). We used an initial value of 0.001 for the learning rate which decreased by a factor of 10 each time the validation loss stagnated after a while and took the model with the lowest loss for validation. In total, 150,852 anterior x-ray images were added to the database, of which 87,020 contained one or more diseases. As for the remaining 63,832 photos, they were normal cases. For the classification process, we randomly divided the data set into three separate groups: training (70%), validation (10%), and testing (20%). In training and validation, we modified the model more precisely using Stochastic Gradient Descent. In our experiments, we only reported the results we obtained in the eight nosologies in the test data. Next, we present the results obtained from our experiments and compare them with other similar methods, namely, the models presented by Wang et al. [11]. We trained our CNN to classify 8 chest diseases contained in the Chest X-ray 8 dataset. In most cases, the models converged quickly, and due to the low complexity of our CNN architecture model, which needed less than 3000 iterations or epochs and obtained Better results than the basic research mentioned in [11]. The graphs in Figure 7 show the accuracy obtained from the validation data in 4 of the eight diseases, the red line represents the testing of our model using the original, unenhanced data, and the blue line shows the results in the cohort. Adjusted data, with increasing data. The accuracy obtained with the validation data for the eight classes is shown in Table 2, in both cases, the original data was not augmented, and the data was augmented with high-quality images in the Chest X-ray 8 dataset. As mentioned above, we can observe significant improvement in all eight categories, which is especially surprising given that our convolutional network is simpler.

**Table 5.2:** Result.

| Label | Precision | Recall | f1-score |
|---|---|---|---|
| Normal | 0.98 | 0.87 | 0.92 |
| Covid-19 | 0.96 | 0.95 | 0.96 |
| Pneumonia | 0.95 | 0.99 | 0.97 |
| Accuracy | | | 0.96 |

| Label | Precision | Recall | f1-score |
|---|---|---|---|
| Normal | 1.00 | 0.97 | 0.99 |
| Covid-19 | 1.00 | 1.00 | 1.00 |
| Pneumonia | 0.99 | 1.00 | 1.00 |
| Accuracy | | | 0.99 |

We get very good results by using a relatively simple AlexNet-inspired convolutional network, as shown in Table 5.1. The authors [11] tested 4 different networks with ChestX-ray8: AlexNet, GoogLeNet, VGG-Net-16, and RestNet50. Better results were obtained with ResNet50, except for "Mass" where AlexNet got the best results. When we compare our model with the results obtained in AlexNet and ResNet50, we see that the results are much improved in all cases of the previous case, except for 'effusion' and better results in 'cardiomegaly', 'infiltration', 'mass' (a particularly difficult category) and "Pneumonia" with the second. We believe that increasing data quality has a very important role in achieving these results, and going forward, we want to check whether the groups that did not get the best results can be improved using other network models or can be improved by modifying the filtering process. As we discussed in the background in this article, the presented data augmentation approach based on image quality yielded very interesting results with Chest X-ray 8 data. To our knowledge, this approach has not been reported in the literature and we consider this to be the main contribution of this job. As mentioned in Section 3.1, we used a simple convolutional network architecture, mainly due to the availability of the hardware we had. This limitation also led us to use smaller images than those reported in the literature (224 × 224, as opposed to 1024 × 1024 images as reported in [11] and other works). However, the results in the previous subsection are very good and promising and we believe that our data augmentation approach can be tested using other, more complex convolutional networks, such as ResNet-50, a line that we can explore in the future using tools like Google Collab or Kaggle Kernels.

# 6. CONCLUSION

Machine learning in medical diagnostics is becoming an even more exciting and applicable field. as it is known the Chest films are the most common type of radiological study in the world, and it is a difficult task, especially for the multiple classifications of medical diagnosis. The chest film, which collects about 45% of all radiological studies, has achieved worldwide popularity and is a low-cost screening tool for many diseases including lung cancer, pneumonia, and tuberculosis. However, the paucity of publicly available medical data has hampered the development of solutions based on deep learning in this field. Furthermore, preprocessing on datasets such as ImageNet or COCO can introduce unintended biases which may be unacceptable in clinical settings, and this makes deep learning more challenging in this field. For example, most clinical settings require models that can accurately predict multiple diagnoses. This turns a lot of medical problems into multiple classification problems, where a large number of results can be for just one image, which can be ambiguous or poorly defined and that image is likely to be inconsistently categorized. These problems are revealed due to the introduction of the chest x-ray data set in 2017, where previously available data sets, such as Open-I, were considered too small. Since then, several studies have progressively improved outcomes when diagnosing chest diseases based on this data set. However, these works do not take into account the image quality of the training examples, which had poor results in previous work, as we demonstrated empirically here using a data augmentation approach with high-quality images. The results of our experiments were able to match the results achieved in 4 diseases and exceed them in 4 other diseases, using a smaller network. We plan to conduct more complete studies where other, more complex networks such as ResNet-50 are evaluated to validate our results.

# REFERENCES


[1] Ooi GC, Khong PL, Müller NL, Yiu WC, Zhou LJ, Ho JC, Lam B, Nicolaou S, Tsang KW (2004) severe acute respiratory syndrome: temporal lung changes at thin-section CT in 30 patients. Radiology 230(3):836–844

[2] Wong KT, Antonio GE, Hui DS, Lee N, Yuen EH, Wu A, Leung CB, Rainer TH, Cameron P, Chung SS, Sung JJ (2003) Severe acute respiratory syndrome: radiographic appearances and pattern of progression in 138 patients. Radiology 228(2):401–406.

[3] Xie X, Li X, Wan S, Gong Y (2006) Mining x-ray images of SARS patients. In: Data Mining. Springer, Berlin, pp 282–294.

[4] Huang C, Wang Y, Li X, Ren L, Zhao J, Hu Y, Zhang L, Fan G, Xu J, Gu X, Cheng Z (2020) Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. Lancet 395(10223):497– 506.

[5] Wang Y, Hu M, Li Q, Zhang XP, Zhai G, Yao N (2020) Abnormal respiratory patterns classifier may contribute to large-scale screening of people infected with COVID-19 in an accurate and unobtrusive manner. arXiv preprint arXiv:2002.05534.

[6] Koo HJ, Lim S, Choe J, Choi SH, Sung H, Do KH (2018) Radiographic and CT features of viral pneumonia. Radiographics 38(3):719–739 7. Li L, Qin L, Xu Z, Yin Y, Wang X, Kong B, Bai J, Lu Y, Fang Z, Song Q, Cao K (2020) Artificial intelligence distinguishes covid-19 from community acquired pneumonia on chest ct. Radiology: 200905.

[7] Hansell DM, Bankier AA, MacMahon H, McLoud TC, Muller NL, Remy J (2008) Fleischner society: glossary of terms for thoracic imaging. Radiology 246(3):697–722.

[8] 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.

[9] Apostolopoulos ID, Mpesiana TA (2020) Covid-19: automatic detection from x-ray images utilising transfer learning with convolutional neural networks. Phys Eng Sci Med:1.

[10] Basile C, Combe C, Pizzarelli F, Covic A, Davenport A, Kanbay M, Kirmizis D, Schneditz D, van der Sande F, Mitra S (2020) Recommendations for the prevention, mitigation and containment of the emerging SARS-CoV-2 (COVID-19) pandemic in haemodialysis centres. Nephrol Dialysis Transplantation 35:737– 741.

[11] Roosa K, Lee Y, Luo R, Kirpich A, Rothenberg R, Hyman JM, Yan P, Chowell G (2020) Real-time forecasts of the COVID-19 epidemic in China from February 5th to February 24th, 2020. Infect Dis Modell 5:256–263.

[12] Yan L, Zhang HT, Xiao Y, Wang M, Sun C, Liang J, Li S, Zhang M, Guo Y, Xiao Y, Tang X

(2020) Prediction of criticality in patients with severe Covid-19 infection using three clinical features: a machine learning-based prognostic model with clinical data in Wuhan. medRxiv.

[13] Xu B, Meng X A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19).

[14] Ajlan AM, Ahyad RA, Jamjoom LG, Alharthy A, Madani TA (2014) Middle East respiratory syndrome coronavirus (MERSCoV) infection: chest CT findings. Am J Roentgenol 203(4):782–787.

[15] Kanne JP (2020) Chest CT findings in 2019 novel coronavirus (2019-nCoV) infections from Wuhan, China: key points for the radiologist.

[16] Afshar P, Heidarian S, Naderkhani F, Oikonomou A, Plataniotis KN, Mohammadi A (2020) Covid-caps: a capsule network-based framework for identification of covid-19 cases from x-ray images. arXiv preprint arXiv:2004.02696.

[17] Chowdhury ME, Rahman T, Khandakar A, Mazhar R, Kadir MA, Mahbub ZB, ..., Reaz MBI (2020) Can AI help in screening viral and COVID-19 pneumonia. arXiv preprint arXiv:2003.13145.

[18] Stoecklin SB, Rolland P, Silue Y, Mailles A, Campese C, Simondon A, Mechain M, Meurice L, Nguyen M, Bassi C, Yamani E (2020) First cases of coronavirus disease 2019 1698 R. Jain et al. (COVID-19) in France: surveillance, investigations and control measures. Eurosurveillance 25(6):2000094.

[19] Alqudah AM, Qazan S, Alqudah A (2020) Automated Systems for Detection of COVID-19 using chest X-ray images and lightweight convolutional neural networks.

[20] Yousefpoor, Mohammad Sadegh, and Hamid Barati. "Dynamic key management algorithms in wireless sensor networks: A survey." Computer Communications 134 (2019): 52-69.

[21] Ayan E, Ünver HM (2019) Diagnosis of pneumonia from chest Xray images using deep learning. In 2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT) (pp. 1-5). IEEE.

[22] Bogoch II, Watts A, Thomas-Bachli A, Huber C, Kraemer MU Khan K (2020) Pneumonia of unknown etiology in Wuhan, China: Potential for International Spread Via Commercial Air Travel. J Travel Med.

[23] Chan YH, Zeng YZ, Wu HC, Wu MC, Sun HM (2018) Effective pneumothorax detection for chest X-ray images using local binary pattern and support vector machine. J Healthcare Eng 2018:1–11.

[24] Chung M, Bernheim A, Mei X, Zhang N, Huang M, Zeng X, Cui J, Xu W, Yang Y, Fayad ZA, Jacobi A (2020) CT imaging features of 2019 novel coronavirus (2019-nCoV). Radiology 295(1):202–207.

[25] Cohen JP, Morrison P, Dao L (2020) COVID-19 image data collection. arXiv preprint

arXiv:2003.11597.

[26] Dadário AMV, Paiva JPQ, Chate RC, Machado BS, Szarf G (2020) Regarding artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT. Radiology:201178.

[27] Diao B, Feng Z, Wang C, Wang H, Liu L, Wang C, Wang R, Liu Y, Liu Y, Wang G, Yuan Z (2020) Human kidney is a target for novel severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) infection. medRxiv.

[28] Fu S, Fu X, Song Y, Li M, Pan PH, Tang T, Zhang C, Jiang T, Tan D, Fan X, Sha X (2020) Virologic and clinical characteristics for prognosis of severe COVID-19: a retrospective observational study in Wuhan, China. medRxiv.

[29] Hassani RTJ, Sandali O (2020) Le nouveau coronavirus Covid-19: quels risques ophtalmiques? J Francais D'Ophtalmologie 43(4):291.

[30] [30] Hassanien AE, Mahdy LN, Ezzat KA, Elmousalami HH, Ella HA (2020) Automatic X-ray COVID-19 lung image classification system based on multi-level Thresholding and support vector machine. medRxiv.

[31] Hemdan EED, Shouman MA, Karar ME (2020) Covidx-net: a framework of deep learning classifiers to diagnose covid-19 in xray images. arXiv preprint arXiv:2003.11055.

[32] Ilyas M, Rehman H, Nait-ali A (2020) Detection of Covid-19 from chest X-ray images using artificial intelligence: an early review. arXiv preprint arXiv:2004.05436

[33] Jacobi A, Chung M, Bernheim A, Eber C (2020) Portable chest Xray in coronavirus disease-19 (COVID-19): a pictorial review. Clin Imaging 64:35–42.

[34] Maghdid HS, Asaad AT, Ghafoor KZ, Sadiq AS, Khan MK (2020) Diagnosing COVID-19 pneumonia from X-ray and CT images using deep learning and transfer learning algorithms. arXiv preprint arXiv:2004.00038.

[35] Ozkaya U, Ozturk S, Barstugan M (2020) Coronavirus (COVID19) classification using deep features fusion and ranking technique. arXiv preprint arXiv:2004.03698.

[36] Ozturk T, Talo M, Yildirim EA, Baloglu UB, Yildirim O, Acharya UR (2020) Automated detection of COVID-19 cases using deep neural networks with X-ray images. Comput Biol Med:103792.

[37] Qiu G, Gai Z, Tao Y, Schmitt J, Kullak-Ublick GA, Wang J (2020) Dual-Functional Plasmonic Photothermal Biosensors for Highly Accurate Severe Acute Respiratory Syndrome Coronavirus 2 Detection. ACS nano

[38] Shi H, Han X, Jiang N, Cao Y, Alwalid O, Gu J, Fan Y, Zheng C (2020) Radiological findings from 81 patients with COVID-19 pneumonia in Wuhan, China: a descriptive study. Lancet Infect Dis 20:425–434.

[39] Thejeshwar SS, Chokkareddy C, Eswaran K, Precise Prediction of COVID-19 in Chest X-Ray

Images Using KE Sieve Algorithm.

[40] Durrani M, Haq I u, Kalsoom U, Yousaf A (2020) Chest X-rays findings in COVID 19 patients at a University Teaching Hospital-A descriptive study. Pakistan J Med Sci 36(COVID19-S4):S22.

[41] Yadav SS, Jadhav SM (2019) Deep convolutional neural network based medical image classification for disease diagnosis. J Big Data 6(1):113.

[42] Zhang J, Xie Y, Li Y, Shen C, Xia Y 2020 Covid-19 screening on chest x-ray images using deep learning based anomaly detection. arXiv preprint arXiv:2003.12338

[43] Zhao W, Zhong Z, Xie X, Yu Q, Liu J (2020) Relation between chest CT findings and clinical conditions of coronavirus disease (COVID-19) pneumonia: a multicenter study. Am J Roentgenol 1–6.

[44] Prashant Patel, Chest X-ray (Covid-19 & Pneumonia), Accessed at: https://www.kaggle.com/prashant268/chest-xray-covid19- pneumonia

[45] Wang S, Kang B, Ma J, Zeng X, Xiao M, Guo J, Cai M, Yang J, Li Y, Meng X, Xu B (2020) A deep learning algorithm using CT images to screen for Corona virus disease (COVID-19). MedRxiv

[46] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016). Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2818-2826

[47] Chollet F (2017) Xception: deep learning with depthwise separable convolutions. In:Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1251-1258

[48] Xie S, Girshick R, Dollár P, Tu Z, He K (2017) Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1492-1500

[49] Sokolova M, Japkowicz N, Szpakowicz S (2006) Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. In: Australasian joint conference on artificial intelligence. Springer, Berlin, pp 1015–1021

[50] Parikh R, Mathai A, Parikh S, Sekhar GC, Thomas R (2008) Understanding and using sensitivity,

[51] WHO. Standardization of interpretation of chest radiographs for the diagnosis of pneumonia in children (2001)

[52] Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.J., Shamma, D. A., Bernstein, M., Fei-Fei, L.: Visual genome: Connecting language and vision using crowdsourced dense image annotations (2016)

[53] Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Ball, R.L., Langlotz, C., Shpanskaya, K., Lungren, M. P., Ng, A.Y.: CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning. https://arxiv.org/abs/1711.05225

[54] Demner-Fushman, D., Kohli, M.D., Rosenman, M., Shooshan, B., Sonya, E., Rodriguez, L., Antani, S., Thoma, G.R., McDonald, C.J.: Preparing a collection of radiology examinations for distribution and retrieval. Journal of the American Medical Informatics Association 23 (2), 304–310 (2015)

[55] Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., Summers, R.M.: Chestx-ray8: Hospitalscale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. arXiv preprint arXiv:1705.02315 (2017)

[56] Chen, S.-F., Chen, Y.-C., Yeh, C.-K., Wang, F., Yu-C.: Order-free rnn with visual attention for multi-label classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017)

[57] Yao, L., Poblenz, E., Dagunts, D., Covington, B., Bernard, D., Lyman, K.: Learning to diagnose from scratch by exploiting dependencies among labels. arXiv preprint arXiv:1710.10501 (2017)

[58] Dodge, S., Karam, L.: Understanding how image quality affects deep neural networks. In: Quality of Multimedia Experience (QoMEX), Eighth International Conference on, IEEE, arXiv:1604.04004v2, pp. 1–6 (2016)

[59] Diamond, S., Sitzmann, V., Boyd, S., Wetzstein, G., Heide, F.: Dirty Pixels: Optimizing Image Classification Architectures for Raw Sensor Data. arXiv:1701.06487

## APPENDIX OF ALL THE CODES USED IN THIS THESIS

```
# USAGE
# python build_covid_dataset.py --covid covid-chestxray-dataset --
output dataset/covid

# import the necessary packages
import pandas as pd
import argparse
import shutil
import os

# construct the argument parser and parse the arguments
ap = argparse.ArgumentParser()
ap.add_argument("-c", "--covid", required=True,
	help="path to base directory for COVID-19 dataset")
ap.add_argument("-o", "--output", required=True,
	help="path to directory where 'normal' images will be stored")
args = vars(ap.parse_args())

# construct the path to the metadata CSV file and load it
csvPath = os.path.sep.join([args["covid"], "metadata.csv"])
df = pd.read_csv(csvPath)

# loop over the rows of the COVID-19 data frame
for (i, row) in df.iterrows():
	# if (1) the current case is not COVID-19 or (2) this is not
	# a 'PA' view, then ignore the row
	if row["finding"] != "COVID-19" or row["view"] != "PA":
		continue

	# build the path to the input image file
	imagePath = os.path.sep.join([args["covid"], "images",
		row["filename"]])

	# if the input image file does not exist (there are some errors
in
	# the COVID-19 metadeta file), ignore the row
	if not os.path.exists(imagePath):
		continue

	# extract the filename from the image path and then construct
the
	# path to the copied image file
	filename = row["filename"].split(os.path.sep)[-1]
	outputPath = os.path.sep.join([args["output"], filename])

	# copy the image
	shutil.copy2(imagePath, outputPath)
imds = imageDatastore('dataset', ...
    'IncludeSubfolders',true, ...
    'LabelSource','foldernames');
```

```
[imdsTrain,imdsTest] = splitEachLabel(imds, 0.70,'randomize');

augimdsTrain  = augmentedImageDatastore([224 224], imdsTrain,
'ColorPreprocessing', 'gray2rgb');
augimdsTest  = augmentedImageDatastore([224 224], imdsTest,
'ColorPreprocessing', 'gray2rgb');

%specify the layers in the network
layers = [
    imageInputLayer([224 224 3])

    convolution2dLayer(3,16,'Padding',1)
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2,'Stride',2)

    convolution2dLayer(3,32,'Padding',1)
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2, 'stride', 2)

    fullyConnectedLayer(2)
    softmaxLayer
    classificationLayer];

options = trainingOptions('sgdm', ...
    'MiniBatchSize',32, ...
    'MaxEpochs',4, ...
    'InitialLearnRate',3e-4, ...
    'Shuffle','every-epoch', ...
    'ValidationData',augimdsTest, ...
    'Verbose',false, ...
    'Plots','training-progress');




net = trainNetwork(augimdsTrain,layers,options);

[YPred,probs] = classify(net,augimdsTest);
accuracy = mean(YPred == imdsTest.Labels);



idx = randperm(numel(augimdsTest.Files),4);
figure
for i = 1:4
    subplot(2,2,i)
    I = readimage(imdsTest,idx(i));
    imshow(I)
    label = YPred(idx(i));
```

```
    title(string(label) + ", " + num2str(100*max(probs(idx(i),:)),3)
+ "%");
end

load('net');

image = imread('test1.jpeg');
aug = augmentedImageDatastore([224 224],image, 'ColorPreprocessing',
'gray2rgb' );

% figure;
%imshow(aug);
[YPred,probs] = classify(net,aug)
accuracy = mean(YPred == imdsTest.Labels)



# USAGE
# python sample_kaggle_dataset.py --kaggle chest_xray --output
dataset/normal

# import the necessary packages
from imutils import paths
import argparse
import random
import shutil
import os

# construct the argument parser and parse the arguments
ap = argparse.ArgumentParser()
ap.add_argument("-k", "--kaggle", required=True,
	help="path to base directory of Kaggle X-ray dataset")
ap.add_argument("-o", "--output", required=True,
	help="path to directory where 'normal' images will be stored")
ap.add_argument("-s", "--sample", type=int, default=25,
	help="# of samples to pull from Kaggle dataset")
args = vars(ap.parse_args())

# grab all training image paths from the Kaggle X-ray dataset
basePath = os.path.sep.join([args["kaggle"], "train", "NORMAL"])
imagePaths = list(paths.list_images(args["kaggle"]))

# randomly sample the image paths
random.seed(42)
random.shuffle(imagePaths)
imagePaths = imagePaths[:args["sample"]]

# loop over the image paths
for (i, imagePath) in enumerate(imagePaths):
	# extract the filename from the image path and then construct
the
	# path to the copied image file
```

```
    filename = imagePath.split(os.path.sep)[-1]
    outputPath = os.path.sep.join([args["output"], filename])

    # copy the image
    shutil.copy2(imagePath, outputPath)
```