REPUBLIC OF TÜRKİYE

ALTINBAŞ UNIVERSITY

Institute of Graduate Studies

Electrical and Computer Engineering

IMPROVING PREDICTION OF CHEST INFECTIONS USING MACHINE LEARNING ALGORITHMS FROM X-RAY IMAGES

Karam Sameer Abdulateef AL-ZUBAIR

Master's Thesis

Supervisor

Assoc. Prof. Dr. Oğuz ATA

Istanbul, 2023

IMPROVING PREDICTION OF CHEST INFECTIONS USING MACHINE LEARNING ALGORITHMS FROM X-RAY IMAGES

Karam Sameer Abdulateef AL-ZUBAIR

Electrical and Computer Engineering

Master's Thesis

ALTINBAŞ UNIVERSITY

2023

The thesis titled IMPROVING PREDICTION OF CHEST INFECTIONS USING MACHINE LEARNING ALGORITHMS FROM X-RAY IMAGES prepared by KARAM SAMEER ABDULATEEF AL-ZUBAIR and submitted on 11/04/2023 has been **accepted unanimously** for the degree of Master of Science in Information Technology.

Assoc. Prof. Dr. Oğuz ATA

Supervisor

Thesis Defense Committee Members:

Assoc. Prof. Dr. Oğuz ATA	Department of Software Engineering, Altınbaş University	___________________
Assoc. Prof. Dr. Doğu Çağdaş ATİLLA	Department of Electrical and Electronics Engineering, Altınbaş University	___________________
Prof. Dr. Hasan Hüseyin BALIK	Department of Computer Engineering, İstanbul Aydın University	___________________

I hereby declare that this thesis meets all format and submission requirements of a Master's thesis.

Submission date of the thesis to Institute of Graduate Studies: ____/____/____

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Karam AL-ZUBAIR

Signature

DEDICATION

Firstly, I would like to thank God Almighty for His grace and mercy to complete my studies successfully.

Secondly, I want to thank my parents and my sister for their unwavering support throughout my academic journey. Their encouragement and belief in me have been crucial in helping me achieve my goals. Their constant support and guidance have helped me stay focused on my studies and motivated me to overcome any obstacles that came my way. Their dedication and sacrifices have not gone unnoticed.

Finally, I cannot forget my friends, Aws Al-Shahery and Hameed Farhan, who gave me invaluable support and encouragement throughout my thesis and helped me submit it in the best possible way.

ACKNOWLEDGEMENTS

I would like to thank my supervisor Assoc. Prof. Dr. Oğuz ATA, for all assets and in my research, encourages the importance of constantly challenging yourself to grow. And the happy times we spent with colleagues while studying for a master's degree.

ABSTRACT

IMPROVING PREDICTION OF CHEST INFECTIONS USING MACHINE LEARNING ALGORITHMS FROM X-RAY IMAGES

AL-ZUBAIR, Karam Sameer Abdulateef

M.Sc., Electrical and Computer Engineering, Altınbaş University,

Supervisor: Assoc. Prof. Dr. Oğuz ATA

Date: April / 2023

Pages: 60

This thesis aims to present and develop a deep learning-based computer application that helps related professionals quickly and accurately identify common lung diseases by interpreting chest X-ray images. Convolutional neural network (CNN) is a very efficient technique for image processing. The proposed work has used AlexNet, and this architecture was fine-tuned and used to extract features from CXR images and analyse these features using energy spectral density individually. The four datasets from open source chest X-ray (CXR) images are publicly available, 15153, 6432, 9208, and 312, respectively, of X-ray images. The experimental results in our proposed system using SVM classifier were accurate at 95.6%, 95.4%, 97.2%, and 95.7%, respectively, for these datasets. The method was compared with several studies.

Keywords: Coronavirus, COVID-19, Deep Learning, Chest X-ray, Energy Spectral Density (ESD), Convolutional Neural Network (CNN), Support Vector Machine (SVM).

TABLE OF CONTENTS

Pages

LIST OF TABLES

Pages

LIST OF FIGURES

Pages

ABBREVIATIONS

DL	:	Deep Learning
CNN	:	Convolutional Neural Network
ESD	:	Energy Spectral Density
SVM	:	Support Vector Machine

1. INTRODUCTION

The virus known as COVID-19 is a very contagious disease transmitted from one infected person to another when the droplets are released when the infected person coughs, talks or sneezes. It is also spread by touching certain surfaces bearing the virus and then involuntarily touching the nose and mouth. Its known symptoms are fever, cough, sometimes severe, and difficulty breathing in some. diarrhoea, nausea, fever, headache, sore throat, body aches, fatigue, and loss of taste or smell may also occur. Signs and symptoms can last from a few days to a few weeks or longer, depending on the severity of the disease. Sometimes, the symptoms are hidden for some people, but the virus can pass to another person, while others are more likely to become severely ill. Those most at risk are the elderly and have severe health problems such as Some have kidney failure, others have severe high blood pressure, and others have pneumonia or a weakened immune system. Severe diseases may lead to organ failure and a high rate of mortality [1]. Reverse polymerase chain reaction (RT-PCR) is presently widely used for COVID-19 diagnosis. However, the duration of this scan is long, so image processing is ideal for early diagnoses of COVID-19, such as chest x-rays and computed tomography (CT). Manual verification of each report is initially difficult and requires the intervention of several radiologists [2].The outbreak of COVID-19 started in December 2019 when a previously unknown coronavirus was identified in Wuhan, China. The early stages of the COVID-19 pandemic involve acute respiratory infections [1]. On January 7, the Chinese Preventive Centre for Disease and Control (CDC) identified 2019-nCoV from a swab sample taken from a patient's throat. A rapid progression to acute respiratory syndrome (ARDS), acute respiratory failure, and other severe conditions has been observed in some people. The World Health Organization later named it accordingly. [3]. Artificial intelligence is the general name for systems that try to emulate a particular part of human intelligence. Many complex transactions are quickly resolved thanks to artificial intelligence. With the development of artificial intelligence, attempts have been made to convince computers that humans have discovered machine learning and its applications [4]. Unlike other AI applications, machine learning consists of algorithms that do not require manual interpretation or application rules to simulate human intelligence. Advances in machine learning serve the intended purpose of diagnosing and treating many diseases [5]. Development in machine learning and artificial intelligence automated some workflows, limiting human intervention. Medicine uses machine learning techniques to detect diseases.

As new data is discovered, machine learning becomes very intelligent and helpful. The more data you have, the more likely you will implement an automated system to detect and classify diseases. Many health studies have been executed using machine learning and artificial intelligence. These methods can detect COVID-19 [6]. In this scenario, reducing the burden on the healthcare system is very important through early detection of the disease for proper treatment. In this context, an automated system based on artificial intelligence can contribute to this process.

1.1 AIM OF THIS STUDY

Rapid diagnosis is essential for early and effective treatment of the spread of epidemics. This study and our deep research aim to develop a deep learning-based computer application that helps related professionals quickly and accurately identify common lung diseases by interpreting chest X-ray images. Our method showed remarkable results in the automatic detection of COVID-19 diseases through X-ray image analysis.

2. LITERATURE REVIEW

Wang and Wong [7] employed an existing dataset of a deep learning model, COVID-Net, to detect COVID-19. This was done by utilizing chest X-rays as input and implementing a specialized Convolutional Neural Network (CNN) based on the COVID-Net architecture. They obtained an accuracy rate of 93.3% with images from the COVID-19 category using a total of 13,975 radiographs.

Li et al. [8] proposed that COVID-MobileXpert uses a Deep Neural Network (DNN) that depends on a lightweight mobile app to utilise CXR images to screen corona case radiological trajectory prediction and the transfer through distillation architecture of pre-trained treating physician extracting chest X-ray image attributes from a large number of infected lung X-rays, the chest X-ray image dataset of a COVID patient contains 179 chest X-ray, 179 Pneumonia, and 179 Normal.

Afshar et al. [9] suggested a Capsule Networks for classifying COVID-19 X-ray images of a dataset called COVID-CAPS to develop a deep neural network (DNN) depending on diagnostic solutions and offered an alternative modelling framework to achieve an accuracy of 95.7%.

Ozturk et al. [10] developed a new predictive model to recognise images of COVID-19 chest X-rays with a classifier called the DarkCovidNet deep learning model trained ResNet50 model using 1125 images Consisting of 125 COVID-19, 500 Pneumonia, and 500 No-Findings images got 98.08% accuracy of in two classes and 87.02% in three.

Apostolopoulos and Mpesiana [11] used convolutional neural network architectures to detect coronavirus using 224 COVID-19, 700 Pneumonia including (400 Bacterial and 314 Viral) and healthy cases with suggested transfer learning to extract significant features from x-ray images. The classifier used VGG19 and MobileNet v2, with an accuracy of 97.40% in two classes and 94.72% in three.

Sethy et al. [12] used many models of convolutional neural network (CNN) to detect COVID-19 using support vector machine (SVM) classifier to extract image features. They prove the ResNet50 model performed best in three classes: 127 COVID-19, 127 Pneumonia (63 Bacterial and 64 Viral), and 127 Normal reaching an accuracy of 95.34%.

Abraham et al. [13] used a modified CNN to detect COVID-19, with many pre-trained CNNs extracting deep features from them. They then processed operating the correlation-based FS

(CFS) method and a Bayesian classifier. Their image dataset includes 435 COVID-19 and 497 non-COVID. The accuracy obtained of 91.1579 % with 181.1436 seconds.

Loey et al. [14] implemented a generative adversarial network (GAN) to analyse COVID-19 cases with deep learning. They used AlexNet, GoogleNet, and RestNet18 as pre-trained models. The total data was about 306 images. The first data included 69 COVID-19, 79 Bacterial Pneumonia, 79 Viral Pneumonia and 79 Normal images, the accuracy in Googlenet was 80.56%. However, the second one included 69 COVID-19, 79 Bacterial Pneumonia and 79 Normal images, with AlexNet gaining an accuracy of 85.19%.

Minaee et al. [15] used CNN to discuss many pre-processing techniques to predict COVID-19 images from other diseases. The dataset trained with these models included 184 COVID-19 and 5000 Normal lung imaging cases, and they obtained an accuracy of 90.89% (± 1.9).

Das et al. [16] suggested a Deep convolutional network to detecting automatically COVID-19 positive cases. The final prediction used a weighted average ensembling technique with DenseNet201, ResNet50V2 and InceptionV3 as classifiers, with an accuracy of 91.62 % in 65 seconds during COVID-19 diagnosis using x-ray images.

Brunese et al. [17] proposed a convolutional neural network based on a three-phase deep learning approach using a pre-trained Visual Geometry Group (VGG-16) network for detecting the type of pneumonia diseases and searching for COVID-19 if found. They used a dataset of 3 classes with 6523 images of x-ray, which were 250 COVID-19, 2753 Pneumonia and 3520 Normal. They divided it into two models. The first compares Normal and lung disease patients, then compare a patient with covid lung disease and pneumonia. Last, the accuracy was reported at 97%.

Mahmud et al. [18] proposed utilising a deep CNN to classify COVID-19. This technique, called COVXNet, was used with many changes to enhance feature extraction depending on variable expansion rates with many algorithms on the trained dataset. The data contain 1220 cases, 305 images of COVID-19, 305 Bacterial Pne., 305 Viral Pne., and 305 Normal. The result of this model was 97.4% of accuracy for COVID-19 vs Normal, 89.6% for COVID-19 vs Bacterial Pne., 87.3% for COVID-19 vs nonCOVID-19 Viral Pne., 89.6% for accuracy for all kinds of pneumonia, and 90.3% of accuracy for four classes.

Duran-Lopez et al. [19] proposed algorithms of a new design of five Convolution Neural Network layers (COVID-XNet) to distinguish COVID-19 cases. The training used five-fold cross-validation. The Class Activation Maps (CAM) help analyse features from COVID-

XNet to localise the COVID-19-infected areas of the screened lungs. The used dataset contains 6926 images, and the model achieved 94.43% accuracy in two classes, 2589 COVID-19 and 4337 Normal.

Abbas et al. [20] presented a deep learning model named DeTraC to detect COVID-19 infection. They reduce the feature space in the decomposition layer, which helps to improve the training efficiency of the framework by yielding more sub-classes. They combine the sub-classes in the composition layer, limiting the training time in the transfer layer to reduce the computation costs. With the database 105 COVID-19, 11 Pneumonia and 80 Normal, this work reached an accuracy of 97.35%.

3. MATERIAL AND METHODS

3.1 DEEP LEARNING

Deep learning is machine learning that enables computers to better understand the concepts of the world around them through learning and recording experience. Again, deep learning is a machine learning technique miming the human ability to observe, analyse, learn and make decisions that solve complexities using large amounts of data without error and mastery [21]. Since the 21st century, the concept of deep learning, which is considered a kind of machine learning, has been used in neural networks that can solve complex and complex problems by simulating human neurons. In a neural network, the complexity increases by exceeding the number of hidden layers five. Deep learning is similar to neural networks, and the main differences are the layers number and the "drop" method utilised to pass the hidden layers. This method randomly removes some nodes during training and interferes with memorisation.

Moreover, deep learning architectures rely on neural networks with multiple hidden layers and parameters. Therefore, deep learning requires less pre-processing than artificial neural networks. Deep learning has taken this name to the next level [22]. The different levels of the model under development constitute the level of depth of the model. A deep learning model can contain hundreds of consecutive levels. Machine learning algorithms typically consist of one or two layers. Unlike machine learning, deep learning methods can learn automatically from data codes such as audio, image, text and video instead of mathematically coded rules. The increasing usage of deep learning for recognition dates back to an error in 1990, and this increased the error rate dropped significantly. In recent years, speech recognition research using deep learning has gained momentum. The literature shows that many studies using deep learning methods yield positive results [33]. As the volume of data grows, deep learning plays an essential role in providing predictive analytics solutions for big data. In particular, this accompanies increased computing power and the development of graphics processors [23]. A lot of research has been used in different deep learning architectures to manage various tasks. Today CNNs in medical imaging are the popular deep learning architecture typologies [24].

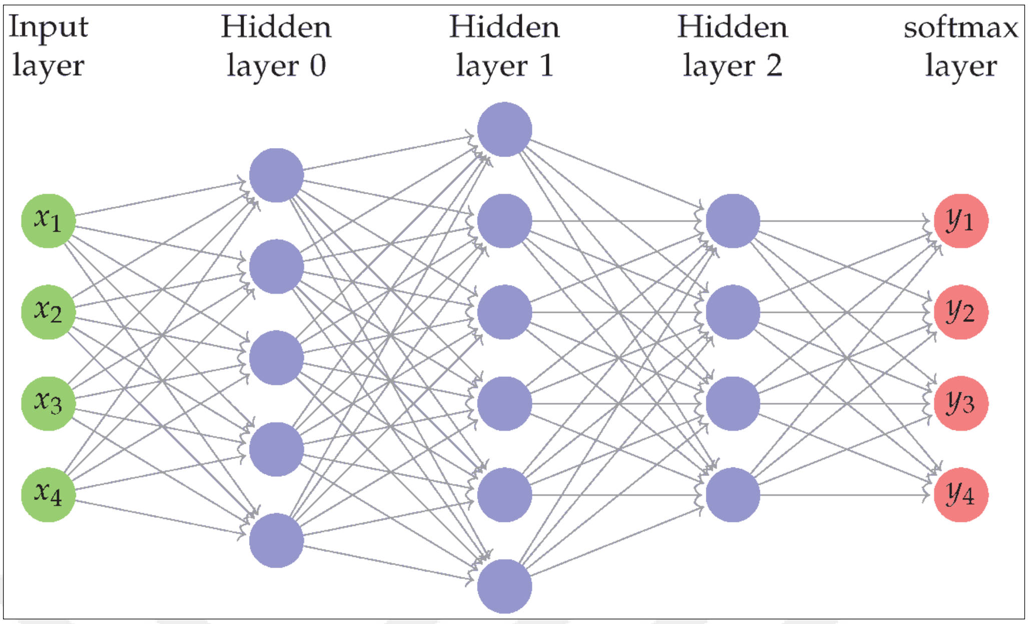

Figure 3.1: Deep Learning [25].

3.2 CONVENTIONAL NEURAL NETWORK (CNN) GENERAL MODEL

The brain's visual system recognises basic patterns in the complex physical world. Animals are more likely to survive and reproduce when they see food or danger signs. However, the representation of objects differs significantly from each other in terms of position, exposure, contrast, background, foreground and many other factors, and it is not easy to distinguish these characteristics from minor image features [26]. In primates, the visual system solves this problem by converting the original image's pixel values into new internal symbols with higher attribute levels [27]. Any image can be modelled with an algorithm that encodes it in a set of non-linear properties [28]. CNN was developed by optimising quantitatively accurate coding patterns in the visual cortex of primates [29]. This network is called a "convolutional neural network" because it uses a mathematical function named convolution. Convolution is a special type of linear operation. At least one of the layers of a convolutional network uses linear operations instead of standard matrix multiplication [30]. This section describes how to use CNNs for classifiers. CNNs have proven to be very effective in practice due to their ability to analyse image properties. It is used on a large scale in many fields, like segmentation and classification. It is more convenient than other image classification algorithms because it requires less preprocessing. Inputs to the neural network are training data obtained from processed image data for CNN operations. The input data class is obtained by processing a hierarchy of convolution, activation, association, smoothing, whole association and classification. Inside are various buildings. CNN is a standard model of artificial neural networks (ANN) consisting of input and output layers merged with several hidden layers. The provided inputs are carried from the neurons and given specific layered functions to produce results [31]. In general, the formulation of the function is represented in figure 3.2.

$$(X, W) = Y \tag{3.1}$$

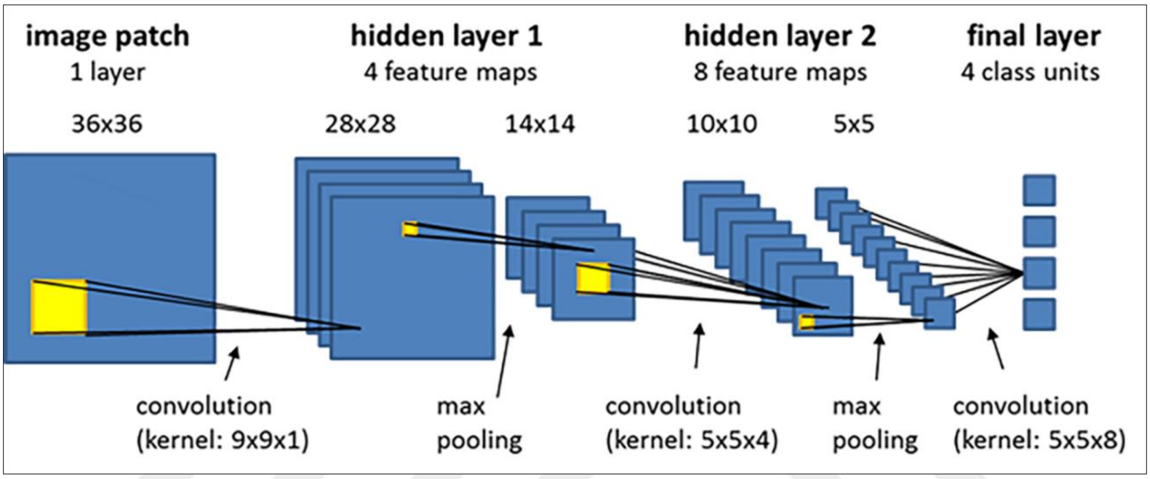

Figure 3.2: CNN [32].

3.2.1 Convolutional Layer

The transform transforms every pixel in the receiving area into a single value. For example, folding an image reduces the image size, combining all the data in the field into one pixel. The final result of the convolutional layer is a vector. Different types of convolution are available, relying on the issue to be solved and the features you want to learn. This is the primary layer of convolutional neural networks. Input images represented by this plane are processed as matrices. In layers, image properties are saved using filters. You can choose a 3x3 or 5x5 filter. The most common convolution type is a 2D convolutional layer called conv2D conv2D. A filter or core in a Conv2D layer places itself on top of the 2D input data and performs the multiplication by element. The result is combined into a single output pixel. The kernel does the same for each position it moves and converts one set of 2D features into another set of 2D features [33]. The Image Feature Map will open. In this exploration, the convolutional layer can be represented in Figure 3.3.

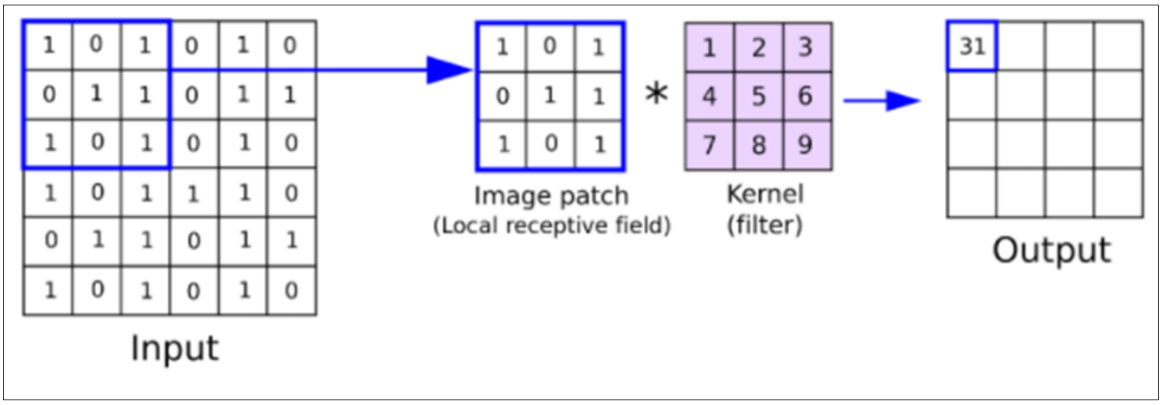

Figure 3.3: Convolutional Layer [34].

3.2.2 Pooling Layer

The general nature of the CNN model is to add multiple convolutional folds and pools one after the other. Grouping levels are operated to reduce feature maps' size, reducing the number of parameters that need to be realised and calculations performed on the network. The repository layer outlines all features in the feature map area that comprise the backlog layer. Therefore, other operations are performed on the joined objects instead of well-placed objects created in the plane of curvature. This makes the form suitable for repositioning elements in the input image. Maximum grouping is a join operation that selects the most oversized item in the feature map area to which the filter is applied. Therefore, after the maximum confluence level, the result is a feature map containing essential features of the previous feature map [35].

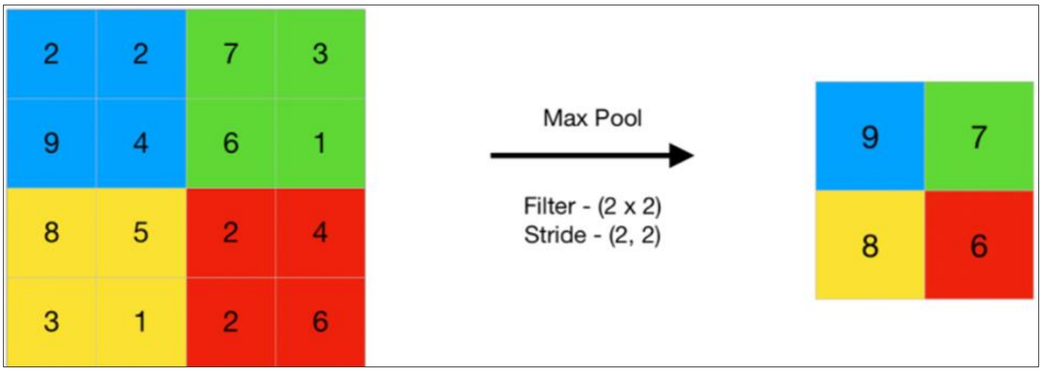

Figure 3.4: Max Pooling [36].

The average criterion uses the entities' average in the feature map's filtered area. Therefore, the largest pool provides essential features for a specific feature map patch, and the average pool is calculated for the features in the patch.

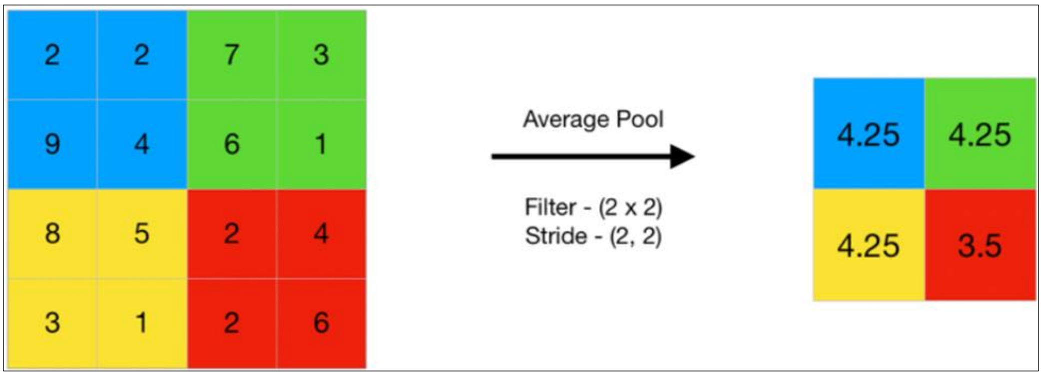

Figure 3.5: Average Pooling [36].

3.2.3 SoftMax Layer

This function is utilised to normalise the output and transform the weighted sum into a probability of becoming 1. Every value in the product of the function is interpreted as a probability of belonging to every class [37]. The function is a combination of several sigmoidal functions. Unlike the sine function used in binary classification, the softmax function takes precedence when there are many classification combinations. This function sets the probability of accumulating those in the input data from 0 to 1. If you build an artificial neural network using the softmax multi-class classification function, the output layer will contain many neurons in the target. The softmax trigger function is primarily used in deep learning architectures at the output level. The softmax function is defined as follows.

$$z_j = \frac{e^{zj}}{\sum_{k=1}^{k} e^{zj}} \; for \; j = 1, \ldots., k \tag{3.2}$$

3.2.4 Activation Function

Activation functions enable us to obtain the net outputs that will be obtained as a result of this process, where operations are performed on the input data in ANN models. Although these functions are rarely linear, they are usually non-linear functions. The reason for choosing non-linear functions is that the model produces successful results in solving a non-linear problem. In addition, functions increase the success of the network but also affect its speed. In cases where we do not implement an activation function in our model, our output signal becomes a simple function. These simple functions are also single-degree polynomials. In this case, it prevents the feature extraction processes of the network, especially in deep learning networks. For this, a non-linear activation function is selected to ensure that the network achieves successful results. Below are the activation functions used in ANN models, especially deep and transfer learning methods [38].

Sigmoid function: This function is the considerably used activation function. The output value is between 0 and 1. It equals 0 if the result is less than zero and one if the value is greater than 0. The response towards the endpoints of the function is reduced, resulting in a smooth gradient. At these points, the derivative values approach 0, and as a result, the gradient disappears. As a result, the network's responsiveness to learning decreases and the time it takes to obtain an accurate prediction increases. In networks where this situation occurs, the hyperbolic tangent function can be chosen instead of this function [39].

$$\mathrm{f(x)} = \frac{1}{1 + e^{-x}} \tag{3.3}$$

Hyperbolic tangent (tanh) function: Although used as an alternative to the sigmoid function, it has a similar structure. However, in this function, the values range from -1 to +1. This range ensures that the function is 0-centred and that the inputs with positive and negative values are modelled more strongly.

$$\mathrm{f(x)} = \frac{1 - e^{-2x}}{1 + e^{-x}} \tag{3.4}$$

ReLu (Rectified Linear Unit) function: The most preferred function to be employed after the convolution layer. Although it has a linear structure, it is preferred because it gives successful results in non-linear problems. The fact that exponential functions are not preferred while performing calculations makes it more advantageous in terms of cost compared to other functions. While the transactions are taking place, the positive values are taken precisely, and the negative values are set to 0. Since the inputs approach zero due to their linear structure, the neuron in the network losses occurs [40].

$$f(x) = Maks(0, x) \tag{3.5}$$

Leaky ReLu (Leaky ReLu) function: It was developed to get rid of the problem of loss of neurons, which is the disadvantage of the ReLu function. It does this using a slight slope instead of 0 in negative fields, which can also produce inconsistent results with negative values [41].

$$f(x) = Maks(a * x, x) \tag{3.6}$$

3.2.5 Transfer Learning

Transfer learning saves time by optimising performance, while transferring knowledge increases learning efficiency and reduces the effort required to memorise information [42]. The given an idea of how the results will be conveyed according to your project. The purpose of transferring learning is to use the information from the source activity to develop learning skills for the target activity. One of the challenges of transfer learning is to create positive transfers and avoid negative transfers between related tasks. There is a negative report because the relationship between source and target assets is weak [43]. That is why it is essential to recognise and reject misinformation when reviewing your target business. In this case, the performance of transfer learning is at least as good as without it. In my research [44], this leads to the idea of examining transfer learning and avoiding passive transfer, which is the development of a new learning task based on transferring knowledge from previously learned related tasks [45]. Transferring training examples, including computer vision, produces good results in object classification [46]. Fortunately, data loss and overload occur in some methods to avoid neural network overfitting [47]. Vehicles, trucks, bicycles, etc. If it is desired to perform the classification with convolutional neural networks, CNN should be trained by training data from these classes. In transfer learning, there is no need to train for defined objects, Cats, dogs, etc., obtained from the outside world. Pictures belonging to classes can be easily classified. For different applications, weights and biases are updated by changing particular layers of the models. It is also seen in the studies that deep learning networks give better results with large datasets. The more data, the higher the learning rate. With the development of hardware features used for computation, the use of deep learning is becoming common. The increment used for the hidden layers in artificial neural networks used in deep learning and deepened networks require more significant memory. At the same time, faster computers must be used [48].

3.2.5.1 Pre-trained model

Models are proposed yearly in Imagenet dataset competitions, and these models have significant success in the field with substantial datasets. The aim here is to use the weights of the models whose training has been completed with large datasets and whose success has been proven. For example, the AlexNet network, which is the winner in the Imagenet competition, has a multi-layer structure and takes days to train even on GPU-supported computers. This network is offered with open access for use after training. It is the most preferred transfer learning approach today. Researchers can then reach the solution more efficiently by using the weight values of this network and revising it according to their own problems [49].

VGG Model: Simonyan and Zisserman proposed VGGNet, members of the VGG (Visual Geometry Group) at Oxford University. The first five significant results were obtained in the ILSVRC2014 competition. It includes more than 14 million entries and 1,000 categories [50]. The model structure is very similar to AlexNet, with the AlexNet filter structure reduced from 11x11 and 5x5 to 3x3. VGGNet has two models, 16 layers and 19 layers. The model is illustrated in figure 3.6

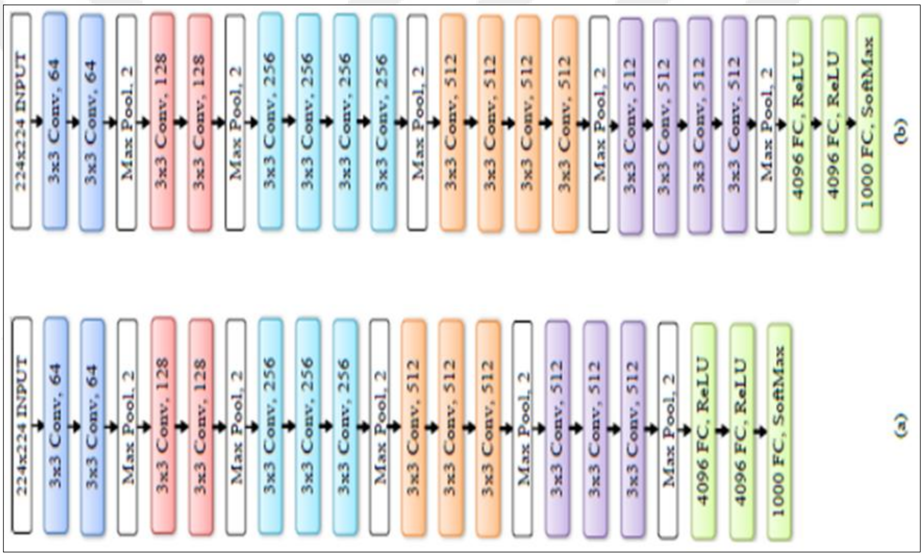

Figure 3.6: VGG Model [51].

ResNet Model: The resulting feature map is analysed by applying filter data of different sizes and numbers. The most important feature distinguishing deep learning architectures from traditional artificial neural network models is that deep network layers can extract map features from data. This structure allows you to analyse a feature map obtained by applying filter data of different sizes and numbers [52].

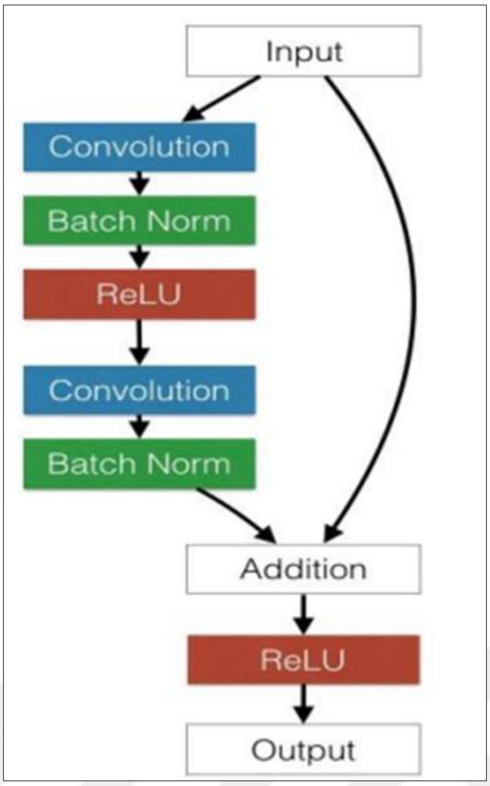

Figure 3.7: ResNet Model [53].

AlexNet: It was developed by Alex Krizhevsky et al. in 2012 and is used on a large scale in transfer learning applications [54]. It is a deep learning model created by training over 1 million images and 1000 various classes (animal, plant, etc.) on the ImageNet database. Image input dimensions are given as 227x227x3. It consists of 25 layers, and approximately 60 million parameters have been calculated. It is constructed using five convolutional layers, 7 ReLu activation functions, three pooling, and finally, three fully connected layers at the end. Using transfer learning methods from AlexNet, smaller-sized datasets can be classified better thanks to previously learned parameters. The first convolutional layer filter is fed with an input image of 96 cores (11 x 11) and 4-pixel steps (227 x 227). The response from the first layer is normalised and pooled before being fed from the second layer. The model is a random gradient descent with group size, momentum coefficient, and weight reduction for training, which uses the optimisation function. AlexNet uses dropout layers to remove overfitting during training [55]. Figure 3.8 shows the details of AlexNet Layers.

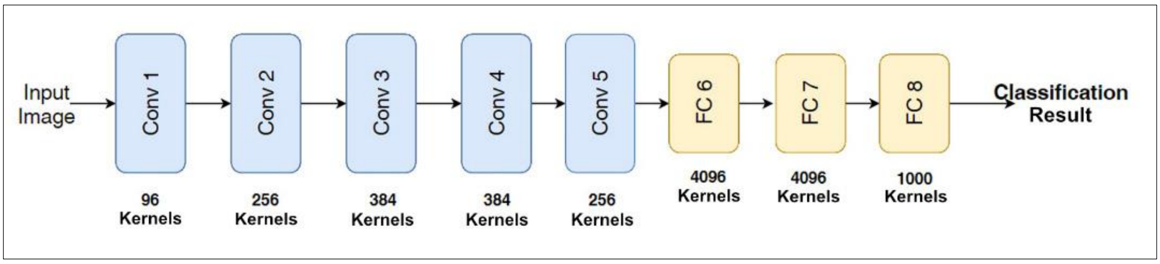

Figure 3.8: AlexNet Architecture [24].

Then, 4096 features were extracted using AlexNet, which these features in the previous studies wired to the classifiers directly. In this study, these features were analysed using energy spectral density. Feature extraction is a process of dimension reduction that, in the original set of raw data, is reduced to greater ease of management packets to process. These datasets are characterized by a large number of variables that require the use of extensive computing power for analysis. The ESD was utilized to decrease the size of the features that were inputted into the classifiers using Equation (3.7):

$$ESD = \int_{-\infty}^{\infty} x(f)^2 df \tag{3.7}$$

Where f refers vector of features between ∞ and ∞ and limits, the ESD represents the output features of the energy spectral density. ESD has always been a true function of frequency, although not negative, and corresponds to the signal energy distribution in the frequency domain. The ESD of signal energy can be interpreted for a given frequency as the energy per unit bandwidth affected by signal around that frequency by the frequency components.

3.2.6 CLASSIFIERS

3.2.6.1 Decision tree (DT)

It is a mapping learning algorithm operated for classifying classes and numerical data. There are predefined goal variables. Define the root node and build a tree on it. When constructing a tree, the algorithm is prepared to perform the end-to-end labelling process. In addition, the algorithm has leaf nodes that support it during the decision stage to achieve any of the top-down goals. You can process large amounts of data quickly. In some cases, a more complex tree should manage log classification. In this case, the decision tree becomes more complex, and it becomes difficult to achieve all the objectives [56]. Excessive branching is another problem with the DT algorithm. To solve this problem, the terminal node is removed from the decision tree. The flowchart of DT is presented in Figure 3.9.

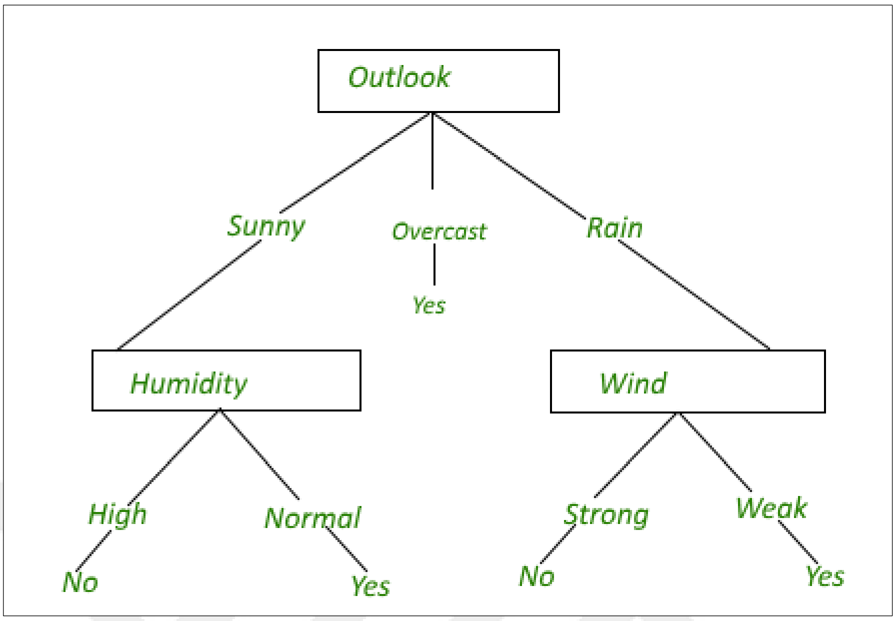

Figure 3.9: DT [57].

3.2.6.2 K-Nearest neighbors (KNN)

It is applied as the classifier for these features and is a supervised learning algorithm. Unlike learning algorithms with other teachers, there are no learning steps. In other words, it determines its neighbors according to a given K value. For this, the K data closest to the new data is selected and added to the selected sample category. The distance of the new data contained in one of the original containers is taken from the data representing the nearest neighbour[58]. Distances are calculated using the Euclidean, Manhattan, and Minkowski functions. Equation (3.8) below shows how these distances are calculated[59].

$$d(p,q) = \sqrt{((q_1 - p_1)^2 + (q_2 - p_2)^2)} \tag{3.8}$$

Then, the flowchart of KNN classifier is presented in figure 3.10:

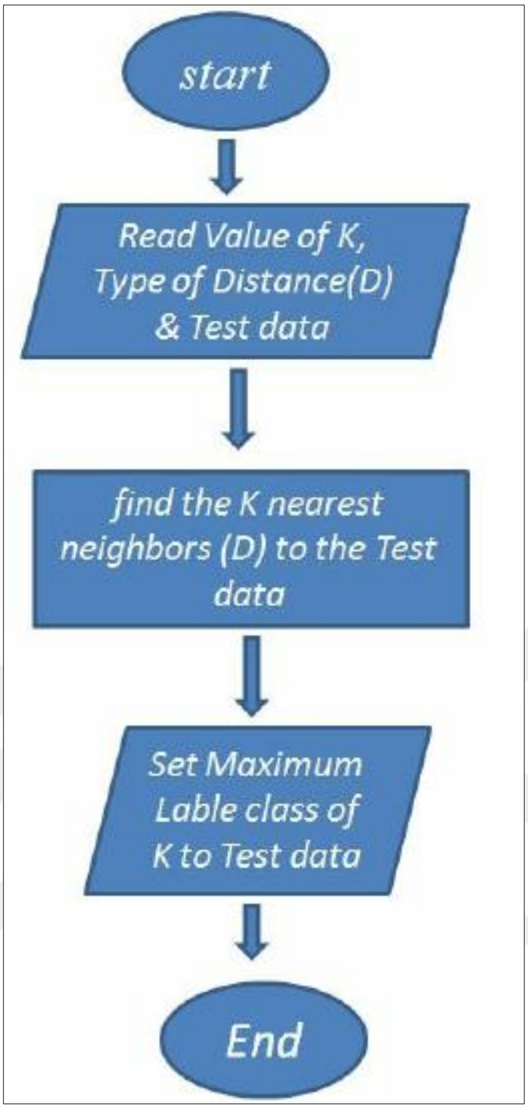

Figure 3.10: KNN [60].

3.2.6.3 Random forest (RF)

It is a supervised learning algorithm used to classify and retreat problems. It is effortless to use. By using DT, a decision forest is created, and this method solves the problem. It generates an RF tree group for it. Multiple DTs are trained to ensure the most accurate classification during the process. In this architecture, the discovery of the root node is performed randomly, which differs from DT. Most of the time, it can lead to good results without using a hyperparameter. It is considered one of the most preferred methods because it gives fast and accurate results even for mixed, incomplete and noisy datasets [61]. The scheme of the random forest is presented in Figure 3.11.

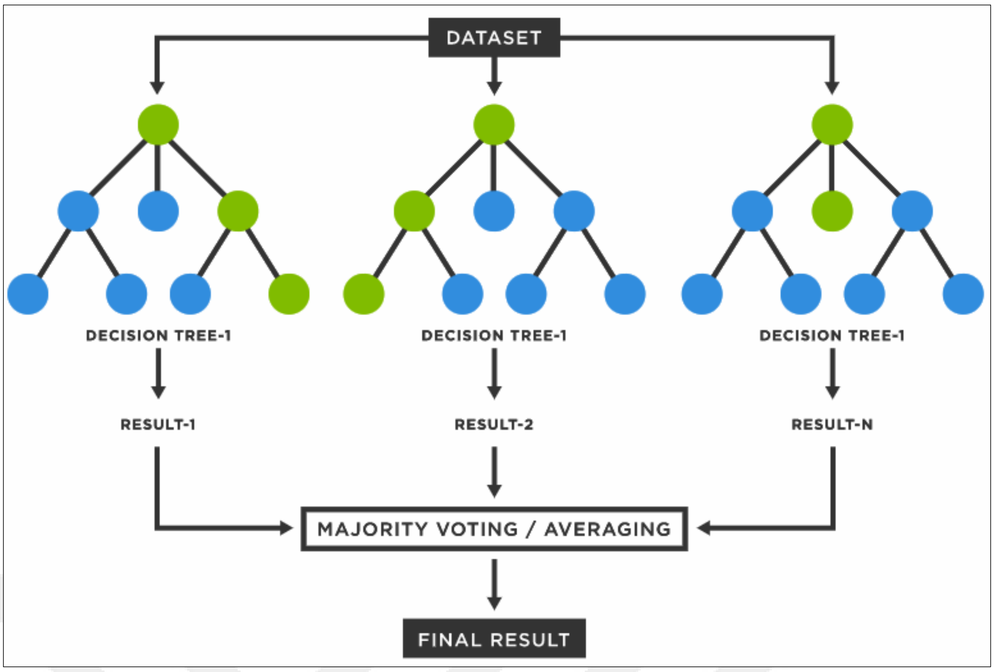

Figure 3.11: RF [62].

3.2.6.4 Support vector machine (SVM)

The final classifier is the most popular machine learning algorithm to classify data into two classes performing well in various fields, especially those used in the medical field and for implementation purposes [63]. The superior objective of this model is to use the ability to distinguish between two specific classes to find the optimal hyperplane that increases the distance of the nearest points in different classes [64], [65]. The line above the point defining which class belongs to the class according to the hyperplane present is called the support vector. The previous section described the classification of SVMs for linearly separable data [66]. However, the data cannot be linearly separated from many real datasets, as shown in figure 3.12. Attempting to classify this data in a linear manner can lead to unsuccessful classification results. When classifying separable non-linear datasets, some errors must occur in order to find the optimal hyperplane. Using non-negative attenuation variables for these datasets, the SVM was designed to account for some of the tradeoffs and drawbacks [67].

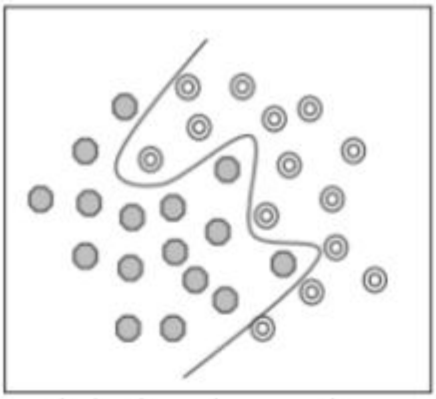

Figure 3.12: Separate Data Linearly [68].

Furthermore, in Figure 3.13, the data will separate in non-linear shape. The classification using SVM consists of two stages linear and non-linear methods.

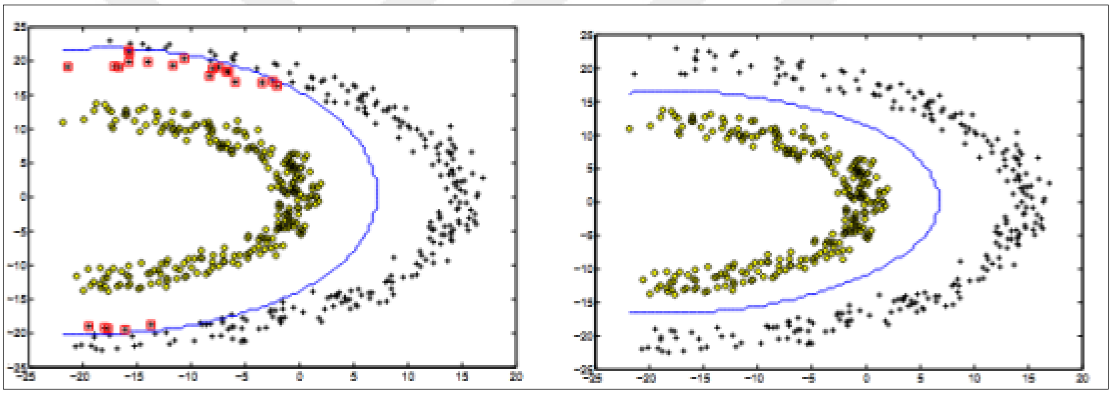

Figure 3.13: Non-linear Classification [69].

3.3 DATASET

Four publicly available trusted sources datasets were obtained and employed in this study, all collected in table 3.1.

a. The first dataset [70] contains 15,153 images (3,616 COVID-19, 10,192 Normal, and 1,345 Viral Pneumonia), taken from:

i. The Valencia Regional Medical Imaging Databank (BIMCV) of medical images [71]

ii. Institute of Diagnostic Radiology, Hannover Medical College, Germany [72]

iii. COVID-19 database of the Italian Department of Interventional and Medical Radiology (SIRM) [73]

iv. The Radiological Society of North America database of Pneumonia disease (RSNA) [74]

v. Chest x-ray images of pneumonia [75]

b. The second dataset [76] with images of 6,432 (576 COVID-19, 1,583 Normal, and 4,273 Pneumonia), taken from:

i. The Shiley Eye Institute, the University of California [77]

ii. Dataset of COVID-19 - Dr. Joseph Cohen "COVID-19 image data collection" [78]

iii. Image Processing and Vision Research Group at the University of Waterloo, Canada [79]. It consists of two sources which are, Dataset of COVID-19 Chest X-ray [80], and Dataset of actual COVID-19 Chest X-ray [81]

c. The third dataset [82] contains 9,208 images (1,281 COVID-19, 3,270 Normal, and 4,657 Pneumonia), taken from:

i. M S Ramaiah Institute of Technology, Concordia University, Indian Institute of Science, PES University

d. The Fourth dataset [83] contains 312 images (132 COVID-19, 90 Normal, and 90 Viral Pneumonia) taken from:

i. Montreal University

This work used images of chest X-rays and corresponding lung datasets. Expert radiologists annotated all cases and illustrated samples of images used for this study in Figure 3.14

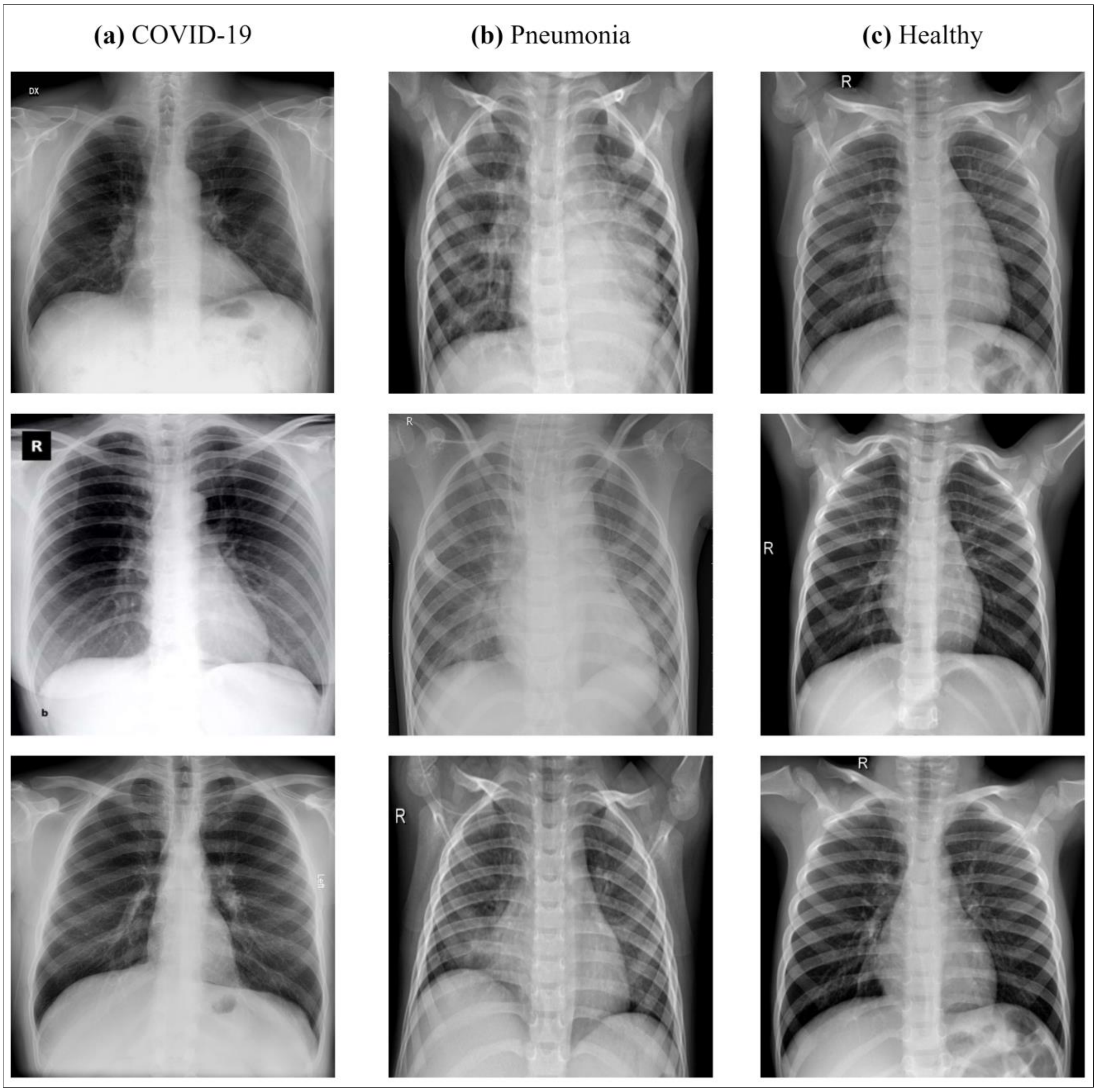

Figure 3.14: Examples of Images are Chest X-rays for Various Classes.

Table 3.1: Data Collection.

Dataset	No. of images			Total number of images	Sources
	COVID-19	Normal	Pneumonia		
1[70]	3,616	10,192	1,345	15,153	- Valencian Region Medical Image Bank (BIMCV) [84] - Hannover Medical School Dataset [85] - SIRM [73] - The Radiological Society of North America database of Pneumonia disease (RSNA) [86] - Chest X-Ray Images (Pneumonia) [75]
2[76]	576	1,583	4,273	6,432	- University of California San Diego [87] - COVID-19 image data collection [88] - University of Waterloo, Canada [79]
3[82]	1,281	3,270	4,657	9,208	- Concordia University, PES University
4[83]	132	90	90	312	- The University of Montreal

4. PROPOSED METHOD

In this study, we developed the method of deep convolutional neural network. Using MATLAB in the first stage, the whole dataset of the chest x-ray is divided in a ratio of 7 to 3 of train and test. The developed system reads the training images, as shown in Algorithm 1.

Algorithm 1: Reading the training images.

```
TraindataFolder = ' D:\karam\Dataset\Train_1';
directoryContent = dir(TraindataFolder);
dirFlags = [directoryContent.isdir];
subFolders = directoryContent(dirFlags);
Traincategories = cell(1,length(subFolders)-2);
for k = 3 : length(subFolders)
   Traincategories{k-2} = [subFolders(k).name];
   subPath=fullfile(TraindataFolder,subFolders(k).name,'\');
end
```

On the other hand, Algorithm 2 shows that the test images are also read using the same code in the previous Algorithm, which is shown below:

Algorithm 2: Reading the testing images.

```
% read the test x-rays
TestdataFolder = ' D:\karam\Dataset\Test_1';
directoryContent = dir(TestdataFolder);
dirFlags = [directoryContent.isdir];
subFolders = directoryContent(dirFlags);
Testcategories = cell(1,length(subFolders)-2);
for k = 3 : length(subFolders)
   Testcategories{k-2} = [subFolders(k).name];
   subPath=fullfile(TestdataFolder,subFolders(k).name,'\');
end
```

Furthermore, AlexNet was used for extracting features of x-ray images. Any size image is converted automatically to 227*227, which is the size that AlexNet can deal with it. The code of this section is presented in Algorithm 3 below:

Algorithm 3: Calling AlexNet.

```
% convert the x-ray images to 227*227 which are the size of AlexNet outputSize=[227 227];
net = alexnet;
net.Layers;
auimdstrain = augmentedImageDatastore(outputSize,imdsTrain);
auimdstest = augmentedImageDatastore(outputSize,imdsTest);
```

In AlexNet, the results were 4096 extracted features. These features in the previous studies were wired to the classifiers directly. This study analysed these features using energy spectral density individually for each train and test data. This function is applied to extracted features from the 1*4096 vector, the power of signal calculates for each of two elements, and each of two elements produces one result, which means reducing the size of the vector from 4096 to 2048. The power of density of the train and the output of the test data are related to the classifier, which reduces computation time and enhances the model's overall accuracy. Then four classifiers, DT, KNN, RF, and SVM, were applied to find the best results for detecting COVID-19. In the first step, a decision tree is applied to classify the extracted features based on their spectral energy density. Algorithm 4 introduces Decision Tree (DT) code as shown below:

Algorithm 4: DT Classifier.

```
treeMd = fitctree(xTrain,YTrain);
treePredicate = predict(treeMd,xTest);
plotconfusion(YTest,treePredicate)
```

The second classifier is K-Nearest Neighbors (KNN), which is also applied to classify the extracted features based on their spectral energy density. The MATLAB code of the KNN presented in Algorithm 5 the show below:

Algorithm 5: KNN Classifier.

```
knnMd = fitcknn(xTrain,YTrain);
knnPredicate = predict(knnMd,xTest);
plotconfusion(YTest,knnPredicate)
```

The Random forest (RF) classifier is the third step and is executed in MATLAB, as shown in Algorithm 6 below:

Algorithm 6: RF Classifier.

```
ensembleMd = fitcensemble(xTrain,YTrain)
ensemblePredicate = predict(ensembleMd,xTest);
plotconfusion(YTest,ensemblePredicate)
```

Finally, the fourth classifier uses SVM in this study for classifying the extracted features. The SVM classifier is the best classification to obtain the best prediction result and get high accuracy in the final evaluation, is shown in Algorithm 7 below:

Algorithm 7: SVM Classifier.

```
svmMd = fitcecoc(xTrain,YTrain);
svmPredicate = predict(svmMd,xTest);
plotconfusion(YTest,svmPredicate)
```

The proposed architecture is shown in Figure 4.1

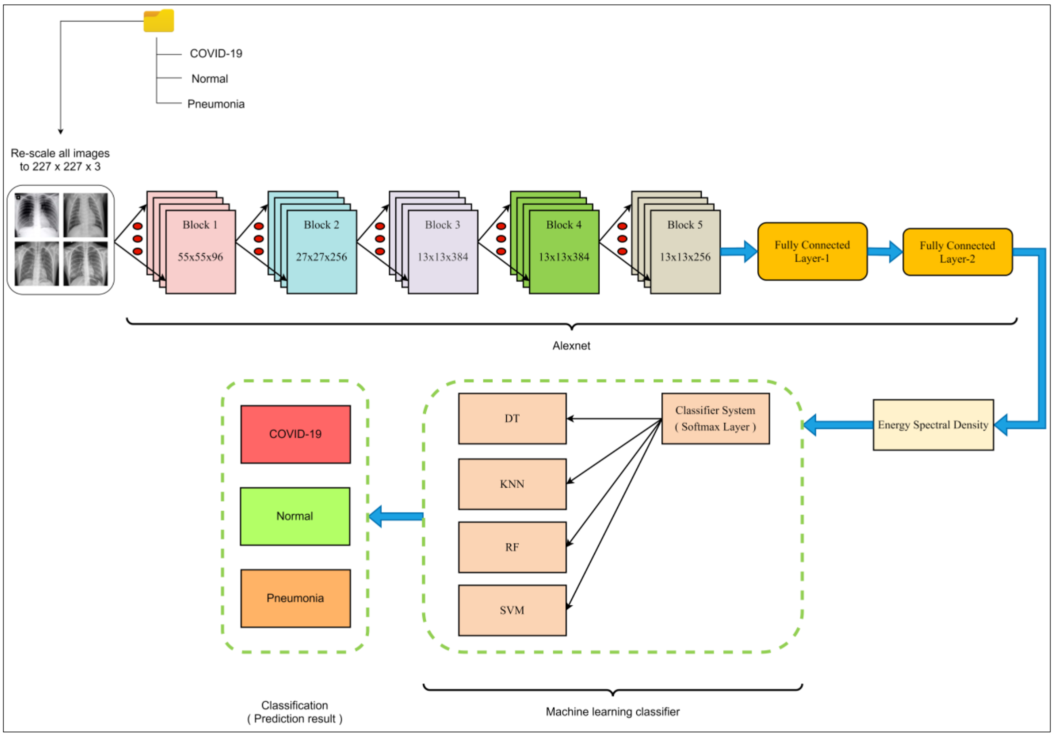

Figure 4.1: Proposed Architecture.

It was also represented as a flowchart of the whole presented method is illustrated in Figure 4.2

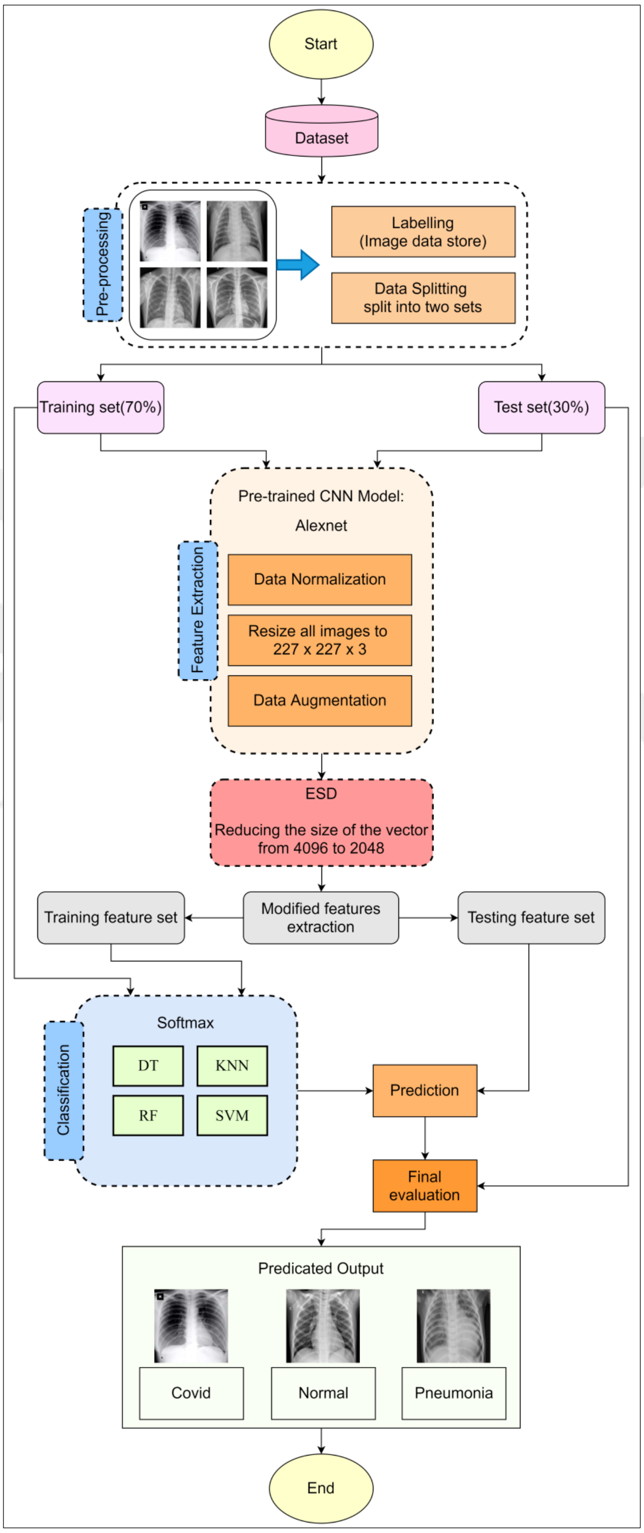

Figure 4.2: Flowchart of the Suggested Technique.

5. EXPERIMENTS AND DISCUSSION

The proposed method in this section has been tested on four X-ray datasets. It then reports the comparison of different machine learning models and demonstrates the benefits of SVM classifiers. It also presents two patient-level case studies and variable significance plots that represent the variables that the model identifies as most important. We point out that our technology works efficiently on large datasets and achieves high results within a short period of time. These four datasets have been used in the confusion matrix of the four classifiers: (DT, KNN, RF, and SVM). Five experiments were applied to each classifier on these same datasets, to mention the minimum result of these experiments obtained and the execution time to prove the efficiency of SVM classifier over the rest of the classifiers. The minimum accuracy and execution time mentioned in seconds are detailed in Table 5.1. Moreover, table 5.2 explains the accuracy of the SVM classifier's five experiments, and the minimum one of the predictions is included in detail as described in the confusion matrix shown below in figures (5.1 – 5.16).

Table 5.1: Comparison Between the Applied Classifiers Used with CNN on a Variety of Datasets, DT, KNN, RF, and SVM

Datasets	DT		KNN		RF		SVM	
	Run Time (Seconds)	Minimum Accuracy (%)	Run Time (Seconds)	Minimum Accuracy (%)	Run Time (Seconds)	Minimum Accuracy (%)	Run Time (Seconds)	Minimum Accuracy (%)
1[70]	280 s	80.6	310 s	91.3	580 s	90.6	400 s	95.6
2[76]	196 s	83.2	202 s	90.4	318 s	92.4	245 s	95.4
3[82]	265 s	86.7	264 s	94.2	462 s	94.1	328 s	97.2
4[83]	43 s	78.7	45 s	88.3	40 s	92.6	43 s	95.7

First dataset:

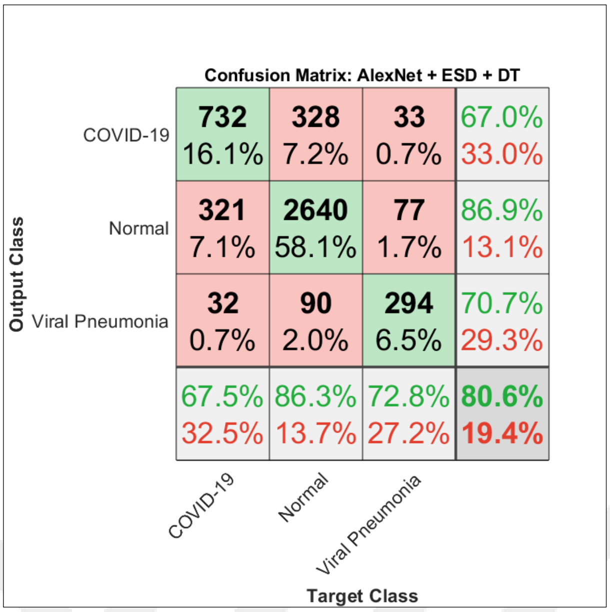

Figure 5.1: CNN + ESD + DT**.**

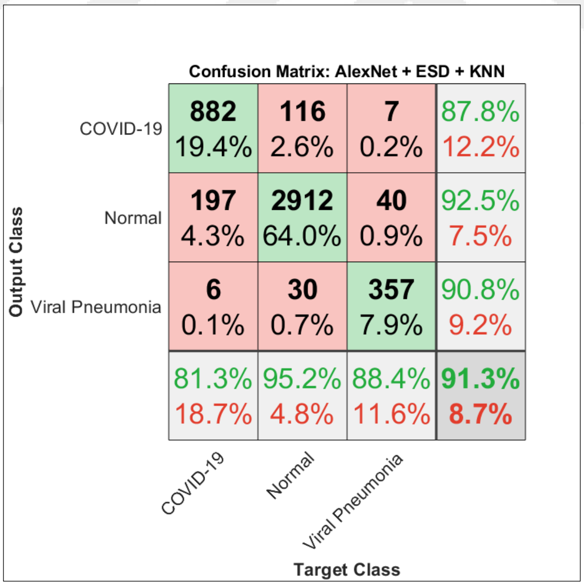

Figure 5.2: CNN + ESD + KNN.

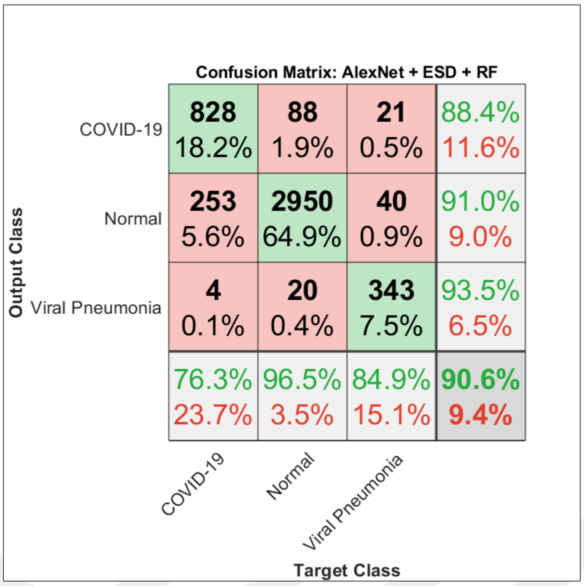

Figure 5.3: CNN + ESD + RF.

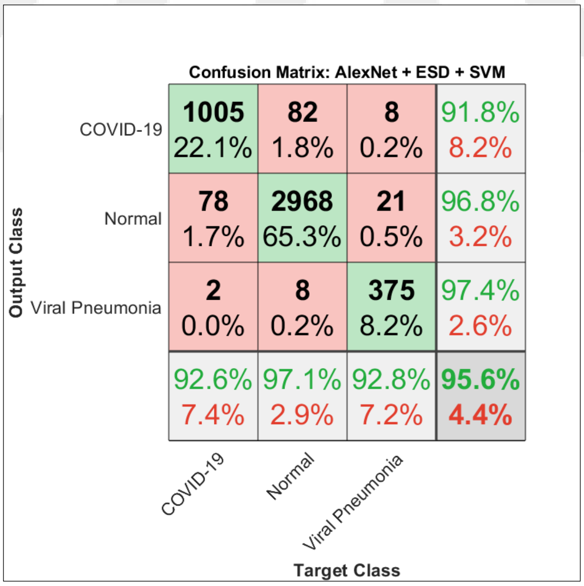

Figure 5.4: CNN + ESD + SVM.

Second Dataset:

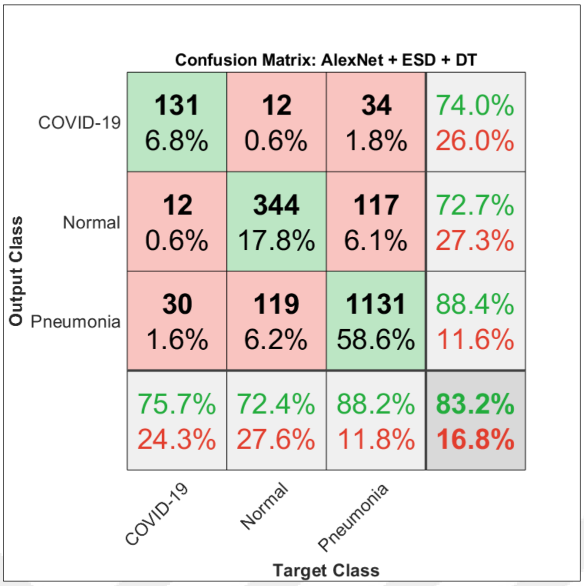

Figure 5.5: CNN + ESD + DT**.**

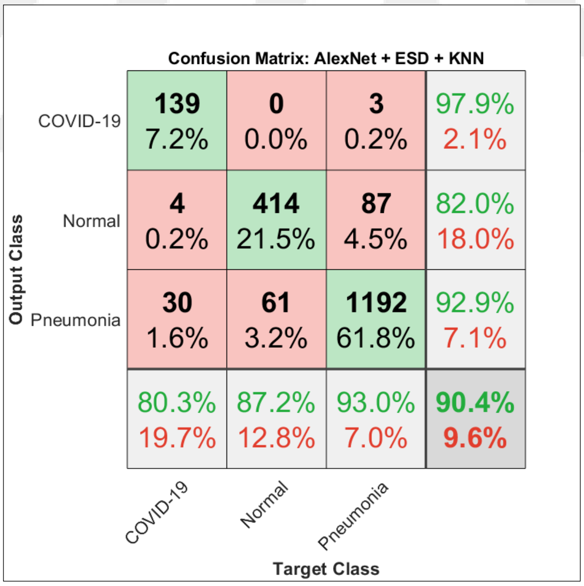

Figure 5.6: CNN + ESD + KNN.

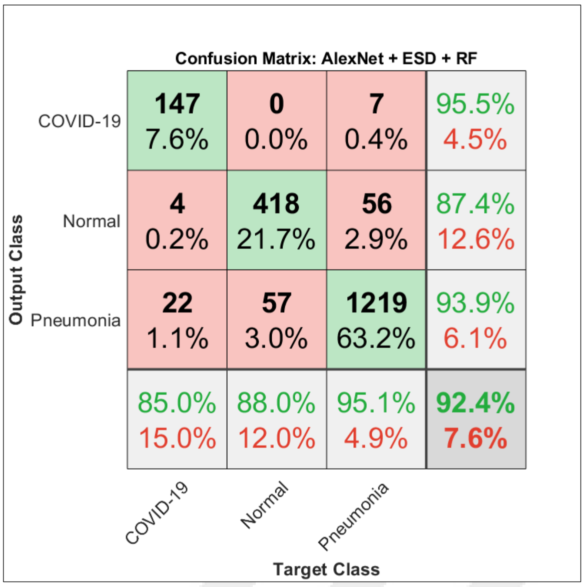

Figure 5.7: CNN + ESD + RF**.**

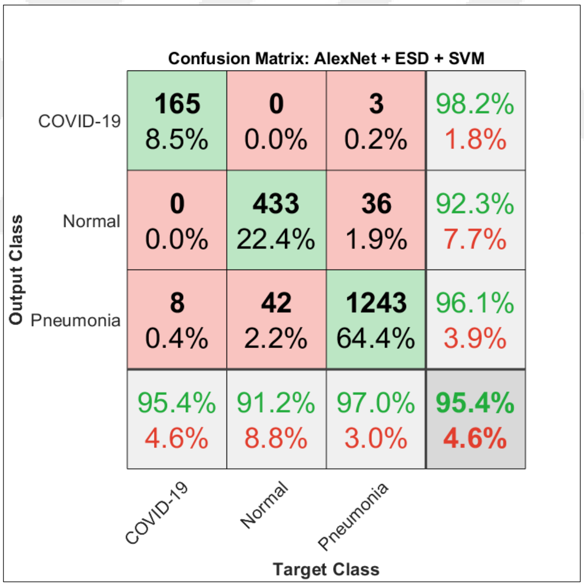

Figure 5.8: CNN + ESD + SVM.

Third dataset:

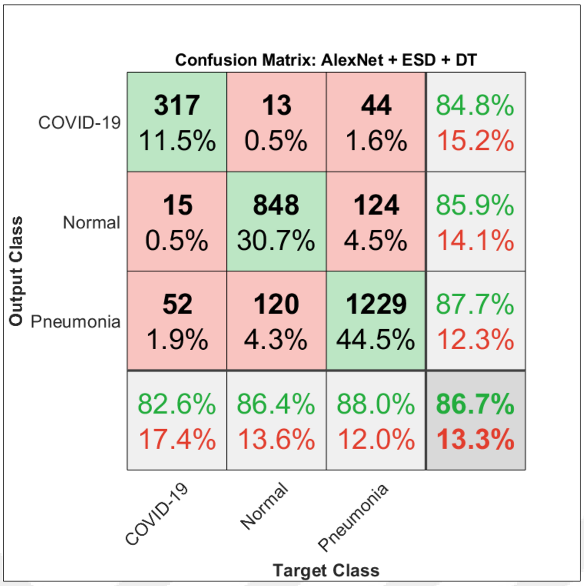

Figure 5.9: CNN + ESD + DT**.**

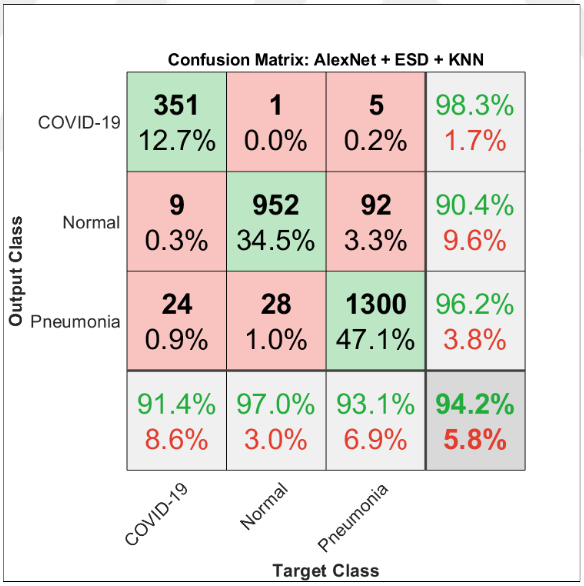

Figure 5.10: CNN + ESD + KNN.

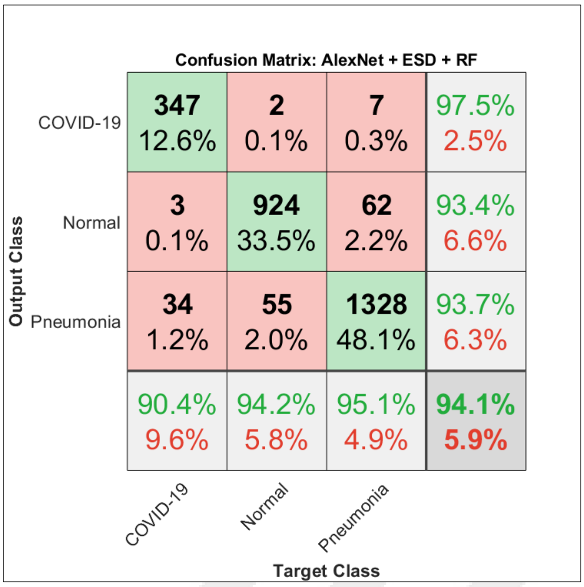

Figure 5.11: CNN + ESD + RF.

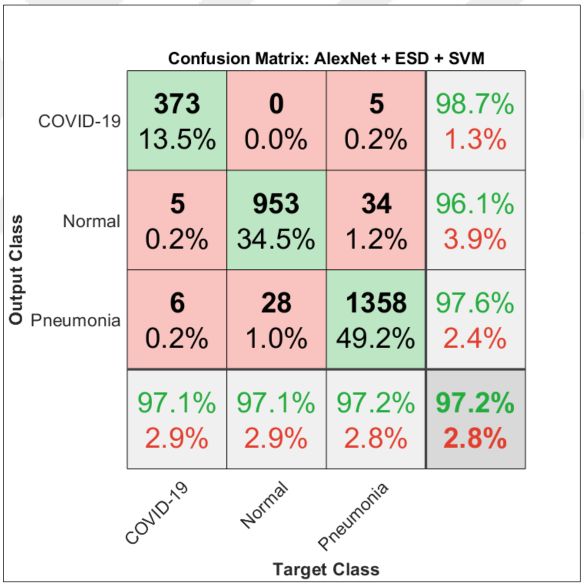

Figure 5.12: CNN + ESD + SVM.

Fourth dataset:

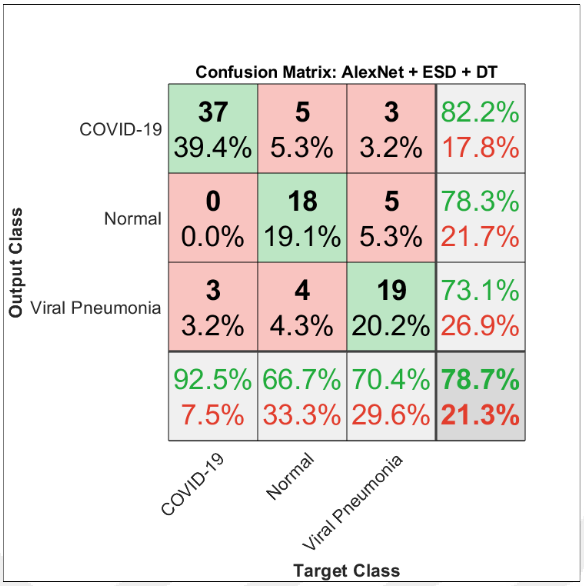

Figure 5.13: CNN + ESD + DT.

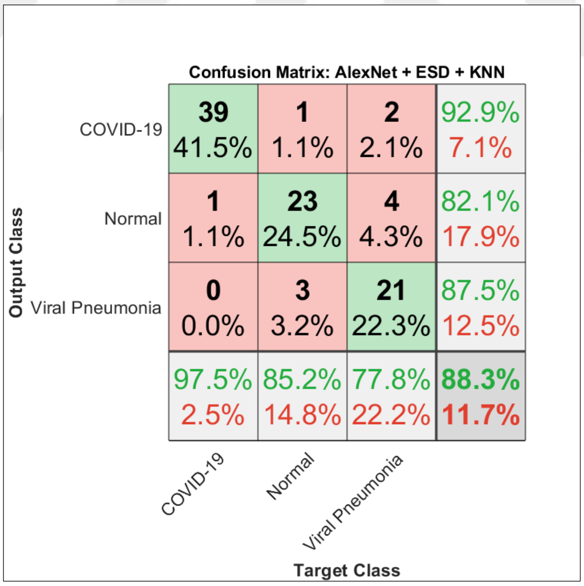

Figure 5.14: CNN + ESD + KNN.

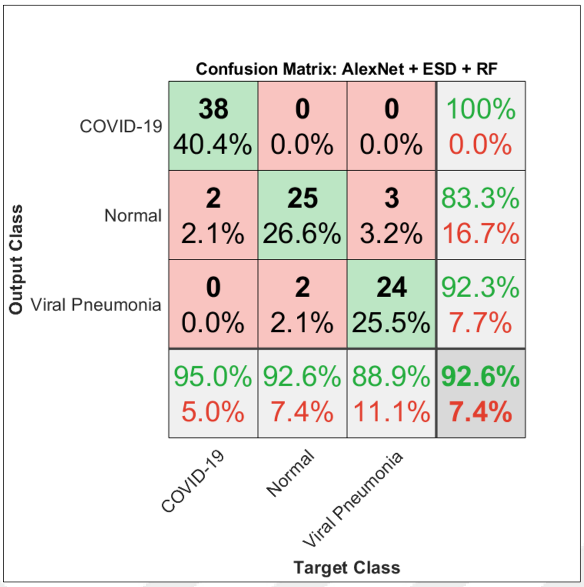

Figure 5.15: CNN + ESD + RF.

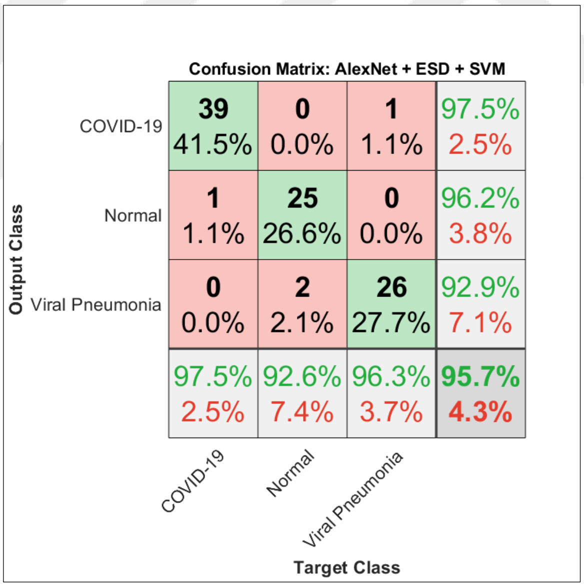

Figure 5.16: CNN + ESD + SVM.

Table 5.2: Applying the SVM Classifier of the Minimum Result of Five Experiments with a Variety of Datasets in the Proposed Method.

Datasets	Predicted labels of the minimum	Actual labels			Performance and Results				
		Actual COVID-19	Actual Normal	Actual Viral Pneumonia	Folds	Accuracy (%)	Minimum (%)	Maximum (%)	Average (%)
1[70]	Predicted COVID-19	1005	78	2	Fold-1	96.3	95.6	96.3	96.04
					Fold-2	96.1			
	Predicted Normal	82	2968	8	Fold-3	96.3			
					Fold-4	95.6			
	Predicted Viral Pneumonia	8	21	375	Fold-5	95.9			
2[76]	Predicted COVID-19	165	0	8	Fold-1	95.9	95.4	96.4	95.92
					Fold-2	95.4			
	Predicted Normal	0	433	42	Fold-3	96.1			
					Fold-4	96.4			
	Predicted Pneumonia	3	36	1243	Fold-5	95.8			
3[82]	Predicted COVID-19	373	5	6	Fold-1	97.7	97.2	97.9	97.5
					Fold-2	97.9			
	Predicted Normal	0	953	28	Fold-3	97.2			
					Fold-4	97.5			
	Predicted Pneumonia	5	34	1358	Fold-5	97.2			
4[83]	Predicted COVID-19	39	0	1	Fold-1	97.9	95.7	97.9	96.8
					Fold-2	95.7			
	Predicted Normal	0	26	1	Fold-3	95.7			
					Fold-4	96.8			
	Predicted Pneumonia	1	1	25	Fold-5	97.9			

Then our method was compared with several studies that applied deep learning techniques with the same datasets they used. Taking the average of five experiments and mentioning the execution time in seconds is shown in the table below.

Table 5.3: The Performance Comparison with Other Existing Deep Learning Methods with the Same Datasets They Used.

Previous study	Methods/classifier	Number of classes	Number of Cases	Run Time	Accuracy (%)	Our Method	
						Accuracy (%) Average	**Run Time**
1[7] 2020	Tailored CNN	3	371 COVID-19 5538 Pneumonia 8066 Normal		93.3	**98.26**	**420 s**
2[8] 2020	Densenet	3	179 COVID-19 179 Pneumonia 179 Normal		88	**96.52**	**54 s**
3[9] 2020	Capsule Network	3	371 COVID-19 5538 Pneumonia 8066 Normal		95.7	**98.26**	**420 s**
4[10] 2020	DarkCovidNet	2	125 COVID-19 500 No-Findings		98.08	**99.9**	**62 s**
4[10] 2020	DarkCovidNet	3	125 COVID-19 500 Pneumonia 500 No-Findings		87.02	**96.46**	**46 s**
5[11] 2020	VGG19	2	224 COVID-19 504 Normal		98.75	**99.52**	**52 s**
5[11] 2020	VGG19	3	224 COVID-19 700 Bacterial Pne. 504 Normal		93.48	**97.38**	**72 s**
5[11] 2020	MobileNet v2	2	224 COVID-19 504 Normal		97.40	**99.52**	**52 s**

Table 5.3: The Performance Comparison with Other Existing Deep Learning Methods with the Same Datasets They Used "Table continued".

5[11] 2020	MobileNet v2	3	224 COVID-19 700 Bacterial Pne. 504 Normal		92.85	**97.38**	**72 s**
5[11] 2020	MobileNet v2	2	224 COVID-19 504 Normal		96.78	**99.52**	**52 s**
5[11] 2020	MobileNet v2	3	224 COVID-19 700 Pneumonia (400 Bacterial and 314 Viral) 504 Normal		94.72	**95.74**	**76 s**
6[12] 2020	ResNet50 + SVM	3	127 COVID-19 127 Pneumonia (63 Bacterial and 64 Viral) 127 Normal		95.34	**96.14**	**38 s**
7[13] 2020	Bayesnet	2	453 COVID-19 497 non-COVID	181.1436 s	91.1579	**99.12**	**62 s**
7[13] 2020	Bayesnet	2	71 COVID-19 7 non-COVID	21.93 s	97.4359	**98.28**	**28 s**
8[14] 2020	Googlenet	4	69 COVID-19 79 Bacterial Pne. 79 Viral Pne. 79 Normal		80.56	**81.3**	**44 s**
8[14] 2020	AlexNet	3	69 COVID-19 79 Bacterial Pne. 79 Normal		85.19	**94.14**	**34 s**
9[15] 2020	ResNet18, ResNet50, SqueezeNet, and DenseNet-121	2	184 COVID-19 5000 Normal		90.89 (± 1.9)	**99.14**	**132 s**

Table 5.3: The Performance Comparison with Other Existing Deep Learning Methods with the Same Datasets They Used "Table continued".

10[16] 2021	New ensemble method combining InceptionV3, Resnet50V2 and Densenet201	2	538 COVID-19 468 non-COVID-19	65 s	91.62	**99.2**	**46 s**
11[17] 2020	Explainable Deep Learning	3	250 COVID-19 2753 Pneumonia 3520 Normal		97	**98.38**	**213 s**
12[18] 2020	CovXNet	2	305 COVID-19 305 Normal		97.4	**99.78**	**53 s**
12[18] 2020	CovXNet	2	305 COVID-19 305 Bacterial Pne.		94.7	**99.34**	**46 s**
12[18] 2020	CovXNet	2	305 COVID-19 305 Viral Pne.		87.3	**96.28**	**48 s**
12[18] 2020	CovXNet	3	305 COVID-19 305 Bacterial Pne. 305 Viral Pne.		89.6	**90.14**	**60 s**
12[18] 2020	CovXNet	4	305 COVID-19 305 Bacterial Pne. 305 Viral Pne. 305 Normal		90.3	**96.08**	**85 s**
13[19] 2020	New Method - COVIDXnet	2	2589 COVID-19 4337 Normal		94.43	**96.08**	**184 s**
14[20] 2021	DeTrac with VGG19	3	105 COVID-19 11 Pneumonia 80 Normal		97.35	**97.96**	**34 s**

5.1 IMPLEMENTATION DETAILS

The proposed transfer learning models are done in MATLAB 2021a and executed using a computer HP ZBook 4th Gen with Processor: Intel (R) Core (TM) i7- 4700MQ CPU @ 2.40GHZ (8 CPUs), 8.00 GB RAM running on an Operating system: Microsoft Windows 10 Professional (64-bit).

6. CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

In this section, we developed a new framework for detecting COVID-19 that uses several classifiers (SVM, KNN, RF, DT) based on a CNN architecture and a tree seed algorithm. In some examples, large functions are the main question of how to work in some machine learning methods. This may result in the model an over-fit or performance degradation during testing. Overfitting the feature selection and process extraction methods was our goal to avoid. In this research, a vital contribution is a combination of genetic algorithms for extracting and selecting CNN features, extracting the active function from the input data of this combination associated with the classifier.

The proposed method applied transfer learning, which assists the method in avoiding overfitting with a low number of training data. The overfitting is the problem that occurs in most deep learning and machine learning studies. Then in this study, transfer learning avoids this problem which Alex Net used as transfer learning.

In each model developed, solutions have been produced to these existing problems, so increasing success rates. In this study, the high success rate of model training with transfer learning, as observed, and results with AlexNet models were obtained in terms of success rate and time. Nevertheless, AlexNet has been noted as these datasets' most successful transfer learning model.

The model receives help from AlexNet in selecting the best features extracted by these features. The 4096 features were reduced to 2084, so it provides the best features with optimal accuracy in detecting COVID-19.

6.2 FUTURE WORK

In future works, it is possible to use other signal processing techniques published in previous years instead of ESD to get new results that can effectively detect COVID-19.

Furthermore, other transfer learning techniques, such as VGG19 and VGG16, can be applied to the same problem. Moreover, the researcher advises using other deep learning techniques instead of convolutional neural network. This led to investigating other deep learning results and comparing these results with convolutional neural network.

Then, other classifiers can also be applied instead of support vector machine to investigate the new results. Furthermore, applying optimisation algorithms to improve the performance of the classifiers to obtain the best results.

REFERENCES

[1] H. Lu, C. W. Stratton, and Y. W. Tang, “Outbreak of pneumonia of unknown etiology in Wuhan, China: The mystery and the miracle,” *J. Med. Virol.*, vol. 92, no. 4, pp. 401–402, Apr. 2020, doi: 10.1002/JMV.25678.

[2] C. R. Dennison Himmelfarb and D. Baptiste, “Coronavirus Disease (COVID-19),” *Journal of Cardiovascular Nursing*, 2020. https://www.who.int/emergencies/diseases/novel-coronavirus-2019 (accessed Nov. 11, 2022).

[3] N. Chen *et al.*, “Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in Wuhan, China: a descriptive study,” *Lancet*, vol. 395, no. 10223, pp. 507–513, Feb. 2020, doi: 10.1016/S0140-6736(20)30211-7.

[4] S. Suganyadevi and V. Seethalakshmi, “CVD-HNet: Classifying Pneumonia and COVID-19 in Chest X-ray Images Using Deep Network,” *Wirel. Pers. Commun.*, vol. 126, no. 4, pp. 3279–3303, Oct. 2022, doi: 10.1007/s11277-022-09864-y.

[5] A. F. Mohammed, S. M. Hashim, and I. K. Jebur, “The diagnosis of COVID-19 in CT images using hybrid machine learning approaches (CNN & SVM),” *Period. Eng. Nat. Sci.*, vol. 10, no. 2, pp. 376–387, Apr. 2022, doi: 10.21533/pen.v10i2.2900.

[6] G. Choy *et al.*, “Current applications and future impact of machine learning in radiology,” *Radiology*, vol. 288, no. 2, pp. 318–328, Aug. 2018, doi: 10.1148/radiol.2018171820.

[7] L. Wang, Z. Q. Lin, and A. Wong, “COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images,” *Sci. Rep.*, vol. 10, no. 1, pp. 1–12, Nov. 2020, doi: 10.1038/s41598-020-76550-z.

[8] X. Li, C. Li, and D. Zhu, “COVID-MobileXpert: On-Device COVID-19 Patient Triage and Follow-up using Chest X-rays,” *Proc. - 2020 IEEE Int. Conf. Bioinforma. Biomed. BIBM 2020*, pp. 1063–1067, Dec. 2020, doi: 10.1109/BIBM49941.2020.9313217.

[9] P. Afshar, S. Heidarian, F. Naderkhani, A. Oikonomou, K. N. Plataniotis, and A.

Mohammadi, “COVID-CAPS: A capsule network-based framework for identification of COVID-19 cases from X-ray images,” *Pattern Recognit. Lett.*, vol. 138, pp. 638–643, Oct. 2020, doi: 10.1016/j.patrec.2020.09.010.

[10] T. Ozturk, M. Talo, E. A. Yildirim, U. B. Baloglu, O. Yildirim, and U. Rajendra Acharya, “Automated detection of COVID-19 cases using deep neural networks with X-ray images,” *Comput. Biol. Med.*, vol. 121, p. 103792, Jun. 2020, doi: 10.1016/j.compbiomed.2020.103792.

[11] I. D. Apostolopoulos and T. A. Mpesiana, “Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks,” *Phys. Eng. Sci. Med.*, vol. 43, no. 2, pp. 635–640, Jun. 2020, doi: 10.1007/s13246-020-00865-4.

[12] P. K. Sethy, S. K. Behera, P. K. Ratha, and P. Biswas, “Detection of coronavirus disease (COVID-19) based on deep features and support vector machine,” *Int. J. Math. Eng. Manag. Sci.*, vol. 5, no. 4, pp. 643–651, 2020, doi: 10.33889/IJMEMS.2020.5.4.052.

[13] B. Abraham and M. S. Nair, “Computer-aided detection of COVID-19 from X-ray images using multi-CNN and Bayesnet classifier,” *Biocybern. Biomed. Eng.*, vol. 40, no. 4, pp. 1436–1445, Oct. 2020, doi: 10.1016/j.bbe.2020.08.005.

[14] M. Loey, F. Smarandache, and N. E. M. Khalifa, “Within the lack of chest COVID-19 X-ray dataset: A novel detection model based on GAN and deep transfer learning,” *Symmetry (Basel).*, vol. 12, no. 4, p. 651, Apr. 2020, doi: 10.3390/SYM12040651.

[15] S. Minaee, R. Kafieh, M. Sonka, S. Yazdani, and G. Jamalipour Soufi, “Deep-COVID: Predicting COVID-19 from chest X-ray images using deep transfer learning,” *Med. Image Anal.*, vol. 65, p. 101794, Oct. 2020, doi: 10.1016/j.media.2020.101794.

[16] A. K. Das, S. Ghosh, S. Thunder, R. Dutta, S. Agarwal, and A. Chakrabarti, “Automatic COVID-19 detection from X-ray images using ensemble learning with convolutional neural network,” *Pattern Anal. Appl.*, vol. 24, no. 3, pp. 1111–1124, Aug. 2021, doi: 10.1007/s10044-021-00970-4.

[17] L. Brunese, F. Mercaldo, A. Reginelli, and A. Santone, “Explainable Deep Learning for Pulmonary Disease and Coronavirus COVID-19 Detection from X-rays,” *Comput. Methods Programs Biomed.*, vol. 196, p. 105608, Nov. 2020, doi: 10.1016/j.cmpb.2020.105608.

[18] T. Mahmud, M. A. Rahman, and S. A. Fattah, “CovXNet: A multi-dilation convolutional neural network for automatic COVID-19 and other pneumonia detection from chest X-ray images with transferable multi-receptive feature optimization,” *Comput. Biol. Med.*, vol. 122, p. 103869, Jul. 2020, doi: 10.1016/j.compbiomed.2020.103869.

[19] L. Duran-Lopez, J. P. Dominguez-Morales, J. Corral-Jaime, S. Vicente-Diaz, and A. Linares-Barranco, “COVID-XNet: A custom deep learning system to diagnose and locate COVID-19 in chest x-ray images,” *Appl. Sci.*, vol. 10, no. 16, p. 5683, Aug. 2020, doi: 10.3390/app10165683.

[20] A. Abbas, M. M. Abdelsamea, and M. M. Gaber, “Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network,” *Appl. Intell.*, vol. 51, no. 2, pp. 854–864, Feb. 2021, doi: 10.1007/s10489-020-01829-7.

[21] M. M. Najafabadi, F. Villanustre, T. M. Khoshgoftaar, N. Seliya, R. Wald, and E. Muharemagic, “Deep learning applications and challenges in big data analytics,” *J. Big Data*, vol. 2, no. 1, pp. 1–21, Feb. 2015, doi: 10.1186/s40537-014-0007-7.

[22] S. Wang, X. Liao, X. Fan, H. Jin, Q. Yao, and Y. Zhang, “Automatically Setting Parameter-Exchanging Interval for Deep Learning,” *Mob. Networks Appl.*, vol. 22, no. 2, pp. 186–194, Apr. 2017, doi: 10.1007/s11036-016-0740-6.

[23] A. Bhavan, M. Sharma, M. Piplani, P. Chauhan, Hitkul, and R. R. Shah, “Deep Learning Approaches for Speech Emotion Recognition,” pp. 259–289, 2020, doi: 10.1007/978-981-15-1216-2_10.

[24] A. M. Karim, H. Kaya, V. Alcan, B. Sen, and I. A. Hadimlioglu, “New Optimized Deep Learning Application for COVID-19 Detection in Chest X-ray Images,” *Symmetry (Basel).*, vol. 14, no. 5, p. 1003, May 2022, doi: 10.3390/sym14051003.

[25] I. Kouretas and V. Paliouras, “Hardware Implementation of a Softmax-Like Function for Deep Learning,” *Technologies*, vol. 8, no. 3, p. 46, Aug. 2020, doi: 10.3390/technologies8030046.

[26] D. Mowen, Y. Munian, and M. Alamaniotis, “Improving Road Safety during Nocturnal Hours by Characterizing Animal Poses Utilizing CNN-Based Analysis of Thermal Images,” *Sustainability*, vol. 14, no. 19, p. 12133, Sep. 2022, doi: 10.3390/su141912133.

[27] W. Byeon, M. Liwicki, and T. M. Breuel, “Scene analysis by mid-level attribute learning using 2D LSTM networks and an application to web-image tagging,” *Pattern Recognit. Lett.*, vol. 63, pp. 23–29, Oct. 2015, doi: 10.1016/j.patrec.2015.06.003.

[28] F. P. An, “Medical Image Classification Algorithm Based on Weight Initialization-Sliding Window Fusion Convolutional Neural Network,” *Complexity*, vol. 2019, 2019, doi: 10.1155/2019/9151670.

[29] N. C. L. Kong, E. Margalit, J. L. Gardner, and A. M. Norcia, “Increasing neural network robustness improves match to macaque V1 eigenspectrum, spatial frequency preference and predictivity,” *PLoS Comput. Biol.*, vol. 18, no. 1, p. e1009739, Jan. 2022, doi: 10.1371/journal.pcbi.1009739.

[30] S. Albawi, T. A. Mohammed, and S. Al-Zawi, “Understanding of a convolutional neural network,” *Proc. 2017 Int. Conf. Eng. Technol. ICET 2017*, vol. 2018-Janua, pp. 1–6, Mar. 2018, doi: 10.1109/ICEngTechnol.2017.8308186.

[31] R. Romero-Zaliz, E. Pérez, F. Jiménez-Molinos, C. Wenger, and J. B. Roldán, “Study of quantized hardware deep neural networks based on resistive switching devices, conventional versus convolutional approaches,” *Electron.*, vol. 10, no. 3, pp. 1–14, Feb. 2021, doi: 10.3390/electronics10030346.

[32] T. eCognition Suite, “Using Deep Learning Models / Convolutional Neural Networks.” https://docs.ecognition.com/eCognition_documentation/User Guide Developer/8 Classification - Deep Learning.htm (accessed Nov. 02, 2022).

[33] M. Sarıgül, B. M. Ozyildirim, and M. Avci, “Differential convolutional neural

network," *Neural Networks*, vol. 116, pp. 279–287, Aug. 2019, doi: 10.1016/j.neunet.2019.04.025.

[34] "Anh H. Reynolds." https://anhreynolds.com/blogs/cnn.html (accessed Nov. 03, 2022).

[35] I. Aizenberg and A. Vasko, "Convolutional neural network with multi-valued neurons," *Proc. 2020 IEEE 3rd Int. Conf. Data Stream Min. Process. DSMP 2020*, pp. 72–77, Aug. 2020, doi: 10.1109/DSMP47368.2020.9204076.

[36] savyakhosla, "CNN | Introduction to Pooling Layer - GeeksforGeeks," *GeeksforGeeks*, 2022. https://www.geeksforgeeks.org/cnn-introduction-to-pooling-layer/ (accessed Nov. 03, 2022).

[37] O. Sharma, "A New Activation Function for Deep Neural Network," *Proc. Int. Conf. Mach. Learn. Big Data, Cloud Parallel Comput. Trends, Prespectives Prospect. Com. 2019*, pp. 84–86, Feb. 2019, doi: 10.1109/COMITCon.2019.8862253.

[38] J. Gu *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognit.*, vol. 77, pp. 354–377, May 2018, doi: 10.1016/j.patcog.2017.10.013.

[39] L. Chen, B. Peng, W. Gan, and Y. Liu, "Plaintext attack on joint transform correlation encryption system by convolutional neural network," *Opt. Express*, vol. 28, no. 19, p. 28154, Sep. 2020, doi: 10.1364/oe.402958.

[40] L. Mou, P. Ghamisi, and X. X. Zhu, "Deep recurrent neural networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3639–3655, Jul. 2017, doi: 10.1109/TGRS.2016.2636241.

[41] S. H. Wang, P. Phillips, Y. Sui, B. Liu, M. Yang, and H. Cheng, "Classification of Alzheimer's Disease Based on Eight-Layer Convolutional Neural Network with Leaky Rectified Linear Unit and Max Pooling," *J. Med. Syst.*, vol. 42, no. 5, pp. 1–11, Mar. 2018, doi: 10.1007/s10916-018-0932-7.

[42] K. M. Hosny, M. A. Kassem, and M. M. Foaud, "Classification of skin lesions using transfer learning and augmentation with Alex-net," *PLoS One*, vol. 14, no. 5, p. e0217293, May 2019, doi: 10.1371/journal.pone.0217293.

[43] W. H. Chen, P. C. Cho, and Y. L. Jiang, "Activity recognition using transfer learning," *Sensors Mater.*, vol. 29, no. 7, pp. 897–904, 2017, doi: 10.18494/SAM.2017.1546.

[44] N. A. Rosli, S. N. H. Sheikh Abdullah, A. N. Zamani, A. Ghazvini, N. S. Md Othman, and N. A. A. A. Muariff Tajuddin, "Comparison Multi Transfer Learning Models for Deep Fake Image Recognizer," *2021 3rd Int. Cyber Resil. Conf. CRC 2021*, Jan. 2021, doi: 10.1109/CRC50527.2021.9392566.

[45] E. F. Ohata *et al.*, "Automatic detection of COVID-19 infection using chest X-ray images through transfer learning," *IEEE/CAA J. Autom. Sin.*, vol. 8, no. 1, pp. 239–248, Jan. 2021, doi: 10.1109/JAS.2020.1003393.

[46] M. Hussain, J. J. Bird, and D. R. Faria, "A study on CNN transfer learning for image classification," *Adv. Intell. Syst. Comput.*, vol. 840, pp. 191–202, 2019, doi: 10.1007/978-3-319-97982-3_16.

[47] M. Grochowski, A. Kwasigroch, and A. Mikołajczyk, "Selected technical issues of deep neural networks for image classification purposes," *Bull. Polish Acad. Sci. Tech. Sci.*, vol. 67, no. 2, pp. 363–376, 2019, doi: 10.24425/bpas.2019.128485.

[48] M. Alkhaleefah, S. C. Ma, Y. L. Chang, B. Huang, P. K. Chittem, and V. P. Achhannagari, "Double-shot transfer learning for breast cancer classification from x-ray images," *Appl. Sci.*, vol. 10, no. 11, p. 3999, Jun. 2020, doi: 10.3390/app10113999.

[49] P. Safayenikoo and I. Akturk, "Weight Update Skipping: Reducing Training Time for Artificial Neural Networks," *IEEE J. Emerg. Sel. Top. Circuits Syst.*, vol. 11, no. 4, pp. 563–574, Dec. 2021, doi: 10.1109/JETCAS.2021.3127907.

[50] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, Sep. 2015, doi: 10.48550/arxiv.1409.1556.

[51] A. Bagaskara and M. Suryanegara, "Evaluation of VGG-16 and VGG-19 Deep Learning Architecture for Classifying Dementia People," *Proc. - 2021 4th Int. Conf. Comput. Informatics Eng. IT-Based Digit. Ind. Innov. Welf. Soc. IC2IE 2021*, pp. 1–

4, 2021, doi: 10.1109/IC2IE53219.2021.9649132.

[52] H. Yoo, S. Han, and K. Chung, “Diagnosis Support Model of Cardiomegaly Based on CNN Using ResNet and Explainable Feature Map,” *IEEE Access*, vol. 9, pp. 55802–55813, 2021, doi: 10.1109/ACCESS.2021.3068597.

[53] “Torch Training and investigating Residual Nets.” http://torch.ch/blog/2016/02/04/resnets.html (accessed Nov. 05, 2022).

[54] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[55] R. Almodfer, S. Xiong, M. Mudhsh, and P. Duan, “Enhancing alexnet for Arabic handwritten words recognition using incremental dropout,” *Proc. - Int. Conf. Tools with Artif. Intell. ICTAI*, vol. 2017-Novem, pp. 663–669, Jun. 2018, doi: 10.1109/ICTAI.2017.00106.

[56] S. R. Safavian and D. Landgrebe, “A Survey of Decision Tree Classifier Methodology,” *IEEE Trans. Syst. Man Cybern.*, vol. 21, no. 3, pp. 660–674, 1991, doi: 10.1109/21.97458.

[57] G. Saloni, “Decision Tree - GeeksforGeeks,” *Geeksforgeeks.org*, 2019. https://www.geeksforgeeks.org/decision-tree/ (accessed Nov. 06, 2022).

[58] Z. Deng, X. Zhu, D. Cheng, M. Zong, and S. Zhang, “Efficient kNN classification algorithm for big data,” *Neurocomputing*, vol. 195, pp. 143–148, Jun. 2016, doi: 10.1016/j.neucom.2015.08.112.

[59] R. Shahid, S. Bertazzon, M. L. Knudtson, and W. A. Ghali, “Comparison of distance measures in spatial analytical modeling for health service planning,” *BMC Health Serv. Res.*, vol. 9, no. 1, pp. 1–14, Nov. 2009, doi: 10.1186/1472-6963-9-200.

[60] M. Bazmara and S. Jafari, “K Nearest Neighbor Algorithm for Finding Soccer Talent,” *J. Basic. Appl. Sci. Res*, vol. 3, no. 4, pp. 981–986, 2013, Accessed: Nov. 07, 2022. [Online]. Available: www.textroad.com

[61] Y. L. Pavlov, “Random forests,” *Random For.*, vol. 45, no. 1, pp. 1–122, Oct. 2019, doi: 10.4324/9781003109396-5.

[62] TIBCO Software Inc., “What is a Random Forest? | TIBCO Software,” 2022. https://www.tibco.com/reference-center/what-is-a-random-forest (accessed Nov. 06, 2022).

[63] A. M. Alqudah, H. M. S. Algharib, A. M. S. Algharib, and H. M. S. Algharib, “Computer aided diagnosis system for automatic two stages classification of breast mass in digital mammogram images,” *Biomed. Eng. - Appl. Basis Commun.*, vol. 31, no. 1, Feb. 2019, doi: 10.4015/S1016237219500078.

[64] Z. Nawel, A. Nabiha, D. Nilanjan, and S. Mokhtar, “Adaptive semi supervised support vector machine semi supervised learning with features cooperation for breast cancer classification,” *J. Med. Imaging Heal. Informatics*, vol. 6, no. 1, pp. 53–62, Feb. 2016, doi: 10.1166/jmihi.2016.1591.

[65] V. Gupta and A. Bhavsar, “Breast Cancer Histopathological Image Classification: Is Magnification Important?,” *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2017-July, pp. 769–776, Aug. 2017, doi: 10.1109/CVPRW.2017.107.

[66] A. Alqudah and A. M. Alqudah, “Sliding Window Based Support Vector Machine System for Classification of Breast Cancer Using Histopathological Microscopic Images,” *IETE J. Res.*, vol. 68, no. 1, pp. 59–67, 2022, doi: 10.1080/03772063.2019.1583610.

[67] A. M. Alqudah, H. Alquran, I. Abu-Qasmieh, and A. Al-Badarneh, “Employing image processing techniques and artificial intelligence for automated eye diagnosis using digital eye fundus images,” *J. Biomimetics, Biomater. Biomed. Eng.*, vol. 39, pp. 40–56, 2018, doi: 10.4028/www.scientific.net/JBBBE.39.40.

[68] Ö. Çelik, “A Research on Machine Learning Methods and Its Applications,” *J. Educ. Technol. Online Learn.*, vol. 1, no. 3, pp. 25–40, Sep. 2018, doi: 10.31681/jetol.457046.

[69] A. Tharwat and A. E. Hassanien, “Quantum-Behaved Particle Swarm Optimization

for Parameter Optimization of Support Vector Machine," *J. Classif.*, vol. 36, no. 3, pp. 576–598, Oct. 2019, doi: 10.1007/s00357-018-9299-1.

[70] Tawsifur Rahman, "COVID-19 Radiography Database," *Kaggle*, 2021. https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database (accessed Sep. 29, 2022).

[71] "BIMCV COVID-19+: a large annotated dataset of RX and CT images from COVID-19 patients | IEEE DataPort." https://ieee-dataport.org/open-access/bimcv-covid-19-large-annotated-dataset-rx-and-ct-images-covid-19-patients (accessed Nov. 11, 2022).

[72] B. C. Winther, Hinrich B. and Laser, Hans and Gerbel, Svetlana and Maschke, Sabine K. and B. Hinrichs, Jan and Vogel-Claussen, Jens and Wacker, Frank K. and Höper, Marius M. and Meyer, "COVID-19 Image Repository," *Figshare*, 2020. https://figshare.com/articles/dataset/COVID-19_Image_Repository/12275009/1 (accessed Nov. 11, 2022).

[73] I. S. of M. and I. Radiology, "Covid-19 Database | Sirm," *Sirm.Org*, 2020. https://www.sirm.org/category/ senza-categoria/covid-19/ (accessed Nov. 11, 2022).

[74] G. Shih *et al.*, "Augmenting the national institutes of health chest radiograph dataset with expert annotations of possible pneumonia," *Radiol. Artif. Intell.*, vol. 1, no. 1, Jan. 2019, doi: 10.1148/ryai.2019180041.

[75] P. Mooney, "Chest X-Ray Images (Pneumonia) | Kaggle," *Kaggle.com*, 2018. https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia (accessed Nov. 11, 2022).

[76] "Chest X-ray (Covid-19 & Pneumonia) | Kaggle." https://www.kaggle.com/datasets/prashant268/chest-xray-covid19-pneumonia (accessed Nov. 11, 2022).

[77] D. S. Kermany *et al.*, "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning," *Cell*, vol. 172, no. 5, pp. 1122-1131.e9, Feb. 2018, doi: 10.1016/j.cell.2018.02.010.

[78] J. P. Cohen, P. Morrison, and L. Dao, "COVID-19 Image Data Collection," Mar. 2020, doi: 10.48550/arxiv.2003.11597.

[79] "agchung · GitHub." https://github.com/agchung (accessed Nov. 10, 2022).

[80] A. Chung, L. Wang, A. Wong, Z. Q. Lin, P. McInnis, and A. Chung, "Figure 1 COVID-19 Chest X-ray Dataset Initiative," *GitHub*, 2020. https://github.com/agchung/Figure1-COVID-chestxray-dataset (accessed Nov. 11, 2022).

[81] L. Wang, A. Wong, Z. Q. Lin, P. McInnis, A. Chung, and H. Gunraj, "GitHub - agchung/Actualmed-COVID-chestxray-dataset: Actualmed COVID-19 Chest X-ray Dataset Initiative," *GitHub*, 2020. https://github.com/agchung/Actualmed-COVID-chestxray-dataset (accessed Nov. 11, 2022).

[82] U. Sait, "Curated Dataset for COVID-19 Posterior-Anterior Chest Radiography Images (X-Rays).," *Mendeley Data*, vol. 3, no. June 2021, pp. 9–10, 2021, doi: 10.17632/9XKHGTS2S6.1.

[83] P. Raikote, "Covid-19 Image Dataset _ Kaggle," 2020. https://www.kaggle.com/pranavraikokte/covid19-image-dataset (accessed Nov. 11, 2022).

[84] M. de la I. Vayá *et al.*, "BIMCV COVID-19+: a large annotated dataset of RX and CT images from COVID-19 patients," 2020. http://arxiv.org/abs/2006.01174 (accessed Nov. 11, 2022).

[85] H. Laser, "GitHub - ml-workgroup/covid-19-image-repository: Anonymized dataset of COVID-19 cases with a focus on radiological imaging. This includes images (x-ray / ct) with extensive metadata, such as admission-, ICU-, laboratory-, and patient master-data.," *GitHub*, 2021. https://github.com/ml-workgroup/covid-19-image-repository (accessed Nov. 11, 2022).

[86] R. S. of N. America, "RSNA Pneumonia Detection Challenge," *Kaggle*, 2018. https://www.kaggle.com/c/rsna-pneumonia-detection-challenge/data (accessed Nov. 11, 2022).

[87] D. Kermany, K. Zhang, and M. Goldbaum, “Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification,” vol. 2, 2018, doi: 10.17632/RSCBJBR9SJ.2.

[88] J. P. Cohen, “ieee8023/covid-chestxray-dataset: We are building an open database of COVID-19 cases with chest X-ray or CT images.,” *GitHub*. https://github.com/ieee8023/covid-chestxray-dataset (accessed Nov. 11, 2022).