ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

CLASSIFICATION OF CHEST X-RAYS BY DIVERGENCE-BASED CONVOLUTIONAL NEURAL NETWORK

M.Sc. THESIS

Muhammed Nur Talha KILIÇ

Department of Electronics and Communication Engineering

Electronics Engineering Programme

APRIL 2022

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

CLASSIFICATION OF CHEST X-RAYS BY DIVERGENCE-BASED CONVOLUTIONAL NEURAL NETWORK

M.Sc. THESIS

Muhammed Nur Talha KILIÇ
(504191217)

Department of Electronics and Communication Engineering

Electronics Engineering Programme

Thesis Advisor: Prof. Dr. Tamer ÖLMEZ

APRIL 2022

ISTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

DİVERJANS TEMELLİ EVRİŞİMSEL AĞ İLE AKCİĞER RÖNTGEN GÖRÜNTÜLERİNİN SINIFLANDIRILMASI

YÜKSEK LİSANS TEZİ

Muhammed Nur Talha KILIÇ
(504191217)

Elektronik ve Haberleşme Mühendisligi Anabilim Dalı

Elektronik Mühendisligi Programı

Tez Danışmanı: Prof. Dr. Tamer ÖLMEZ

NİSAN 2022

Muhammed Nur Talha KILIÇ, a M.Sc. student of İTU Graduate School student ID 504191217, successfully defended the thesis entitled "Classification of Chest X-Rays by Divergence-Based Convolutional Neural Network", which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Prof. Dr. Tamer ÖLMEZ** ....
İstanbul Technical University

Jury Members : **Prof. Dr. Selçuk PAKER** ....
Istanbul Technical University

Prof. Dr. Nizamettin AYDIN ....
Yıldız Technical University

Date of Submission : 22 April 2022
Date of Defense : 13 May 2022

To my family and friends,

FOREWORD

I would like to thank my precious family, friends, who have always supported me and who have always been by my side and I would like to thank my thesis advisor, Prof. Tamer ÖLMEZ, for his invaluable guidance and constant support during the thesis process.

I hope that this thesis, which is the output of the studies I have worked on during my graduate education, will contribute to the literature.

I would like to thank the National Center for High Performance Computing of Turkey (UHeM) for computing resources under grant number 4009822021.

April 2022

Muhammed Nur Talha KILIÇ
(Electronics Engineer)

TABLE OF CONTENTS

Page

ABBREVIATIONS

AI	**:** Artificial Intelligence
GPU	**:** Graphics Processing Unit
CNN	**:** Convolutional Neural Network
FCNN	**:** Fully Connected Neural Network
CXR	**:** Chest X-Ray
WAM	**:** Weighted Arithmetic Mean
CT	**:** Computed Tomography
CLAHE	**:** Contrast Limited Adaptive Histogram Equazation
ReLU	**:** Rectified Linear Unit
WHO	**:** World Health Organization
VOC	**:** Variant of Concern
RNA	**:** Ribonucleic Acid
PTSD	**:** Post-Traumatic Stress Disorder
RGB	**:** Red, Green, Blue
SGD	**:** Stochastic Gradient Descent
UHeM	**:** The National Center for High Performance Computing

SYMBOLS

$\boldsymbol{u}$	**:** Mean
$\boldsymbol{\sigma}$	**:** Variation
$\boldsymbol{w}$	**:** Weight
$\boldsymbol{\ell}$	**:** Learning Rate
γ	**:** Momentum
exp	**:** Exponential
t	**:** Time
p	**:** Probabilty

LIST OF TABLES

Page

LIST OF FIGURES

Page

CLASSIFICATION OF CHEST X-RAYS BY DIVERGENCE-BASED CONVOLUTIONAL NEURAL NETWORK

SUMMARY

The importance of imaging methods in the field of health is increasing day by day with the opportunities provided by technology. Imaging without physical intervention is both more convenient and less costly for the patient and one of the ways to diagnose faster. Diagnosis of diseases and taking action, especially in the early stages of epidemic diseases, are among the most effective methods in the fight against these diseases. Correct treatments are applied against diseases that can be differentiated from each other, thus the patient's recovery is ensured.

Chest radiographs are also one of the most frequently used methods in the field of imaging, and the damage caused by various viruses or bacteria to the lung can be understood and diagnosed with X-ray images. At the same time, it ensures that health systems continue with minimum damage by providing the necessary data for health workers to take precautions in case of an infectious side of the disease.

Covid-19, which entered our lives towards the end of 2019 and caused the death of millions of people in more than 2 years, directly damages the lungs and causes the lungs to not perform their functions. Problems that occur in the lungs by reducing oxygen saturation cause many issues, especially respiratory problems in patients.

Some of the problems seen in patients with suspected Covid-19 occur in the lungs and these changes caused by the disease can be detected by X-rays. However, it is not known to what extent the effects of viruses such as Covid-19, which have been in existence for years but have the capacity to infect people in the near future, and it is not possible to deliver treatment techniques to all parts of the world in a short time. In other words, it is a very long and laborious process to be able to effectively diagnose diseases that we can call new diseases that come into our lives by health workers around the world. For this reason, engineering applications in the field of medical imaging are promising in many respects. For example, AI-assisted engineering applications, which have become very common recently, bring many advantages.

These achievements can be listed as follows:

- Detailed analysis opportunity
- Instant access to developments in the world
- Ability to be easily updated
- Possibility to get results with high accuracy and fast
- To alleviate the burden of healthcare workers
- Cost-reducing contributions with fewer employees

- Reaching areas with low or limited access to the health system
- Reducing the contagious risk of the disease by reducing direct or indirect contact with patients
- Creating systems whose accuracy and reliability are increasing day by day with continuously trained models.
- Opportunity to create models trained with more examples than specialist doctors can see throughout their career

Although the developments mentioned are undeniably positive, the physical hardware needs that come with artificial intelligence supported applications should not be ignored. Datasets with thousands of images prepared for the models to learn the training process bring along the hardware bottleneck. If the appropriate systems are not prepared, model learning takes too long or cannot be done. Considering the pre-trained models, which give very high results in terms of accuracy, the number of parameters is in the millions. It is quite inconvenient that these parameters are constantly updated during training. On the other hand, variables called hyperparameters, whose values are not clear, can only be found by testing in the system. In this case, hundreds of trials should be made until the appropriate model is found and the optimum one should be selected. As a result, while making such applications, issues such as memory, processor power, GPU support appear as parts that need to be solved technically.

In this study, applications on the CNN architecture used in image classification have been carried out.The FCNN layer included in CNN varies from model to model and has many parameters that need to be updated. We thought that changes in this layer might lighten the model and solve the bottleneck problem. With this idea, the FCNN layer was completely removed and the model was tried to be learned only with the feature extractor layer, and a model in which Walsh functions were included was designed. In this way, the number of parameters has been reduced from models that require millions to approximately two hundred fifty thousand, which means one in six hundred when compared to VGG-19 [1]. With the proposed model, X-ray images belonging to 4 different classes were trained and analyzed. While promising with an accuracy rate of 96.15%, it drastically reduced the training time and the need for powerful machines. The obtained outputs were analyzed and the points to be taken into account in future studies were determined.

DİVERJANS TEMELLİ EVRİŞİMSEL AĞ İLE AKCİĞER RÖNTGEN GÖRÜNTÜLERİNİN SINIFLANDIRILMASI

ÖZET

Sağlık alanındaki görüntüleme yöntemlerinin önemi teknolojinin sağladığı imkanlar ile birlikte her geçen gün artmaktadır. Fiziksel müdahale olmadan yapılan görüntülemeler hem hasta açısından daha uygun ve de az maliyetli hem de doktorlar açısından daha hızlı teşhis koymanın yollarından biridir. Özellikle erken dönemde yapılan görüntülemelerde hastalıkların teşhisi ve aksiyonların alınması hastalıklar ile mücadelede etkili yöntemlerin başında gelmektedir. Birbirinden ayrıştırılabilen hastalıklara karşı doğru tedaviler uygulanmakta böylelikle hastanın iyileşmesi sağlanmaktadır.

Medikal görüntüleme son yıllarda oldukça popüler bir alandır. Görüntülerin oluşturulmasında ses, manyetik alan, x ışınları gibi farklı teknikler kullanılmaktadır. X ışınları kullanılarak oluşturulan akciğer röntgenleri de görüntüleme alanında sıklıkla kullanılan yöntemlerden biridir ve çeşitli virüs ya da bakterilerin akciğere verdikleri hasarlar röntgen görüntüleri ile anlaşılabilmekte ve teşhis koyulabilmektedir. Aynı zamanda hastalığın bulaşıcı bir tarafı olması durumunda sağlık çalışanlarının önlem alabilmesi için gerekli verileri sunarak sağlık sistemlerinin minimum hasarla devam etmesini sağlamaktadır.Öte yandan yapılan bu çalışmanın ihtiyacı olan görüntüler de erişilebilirlik açısından daha avantajlı olan x ışınları ile oluşturulan akciğer röntgenleri seçilererek tamamlanmıştır.

2019 senesinin sonlarına doğru hayatımıza giren ve 2 seneden fazla bir sürede milyonlarca insanın ölümüne neden olan Covid-19 da akciğerlere doğrudan zarar vermekte ve de akciğerlerin işlevlerini yerine getirememesine sebep olmaktadır. Hastalardaki oksijen doygunluğunu azaltacak şekilde akciğerlerde neden olduğu problemler hastaların nefes sorunları başta olmak üzere bir çok sorun yaşanmasına neden olmaktadır. Bulaşıcı olmasından dolayı da kısa sürede yayılabilmekte ve sağlık sistemi dahil olmak üzere insanların çalıştığı bir çok sektörde sistemi tıkayıcı şekilde etkiler göstermektedir. Böyle bir durumda tıbbi gereksinimleri karşılanamayan bir çok insan hayatını kaybetmektedir. Bu sebeble mümkün olan en kısa sürede hastalık şüphesi ile gelen insanların arasında Covid-19 virüsüne sahip olanlar ayrıştırılarak izole edilmeli ve hastalığın yayılmasının önüne geçilerek temel sektörlerin işleyişi sürdürülmelidir.Bu da hastalığın teşhisi konusunda çalışmaların ne derece önemli olduğunu gösteren bir başka durumdur.

Covid-19 şüphesi olan hastalarda görülen problemlerin bir kısmı akciğerlerde meydana gelmektedir. Hastalığın neden olduğu değişimler röntgenler aracılığı ile tespit edilebilmektedir. Fakat Covid-19 gibi aslında senelerdir var olan ama insanlara yakın zamanda bulaşarak hasta edebilme kapasitesine sahip virüslerin ne derece etki edeceği tam olarak bilinmemekle birlikte dünyanın her tarafına bu bilgilerin kısa

sürede ulaştırılması da mümkün olmamaktadır. Yani yapılan çalışmaların dünyadaki sağlık çalışanlarına ulaşıp, hayatımıza giren yeni diyebileceğimiz hastalıkların teşhislerini etkili bir şekilde yapabilmek oldukça uzun ve de zahmetli bir süreçtir. Bu sebeple medikal görüntüleme alanında yapılacak mühendislik uygulamaları birçok açıdan umut vaat etmektedir. İnsan gücünden bağımsız oluşu ve sürekli güncellenerek geliştirilebilmesi mühendislik uygulamalarına olan ilgiyi arttırmaktadır. Örnek vermek gerekirse, son zamanlarda kullanımı oldukça yaygınlaşan yapay zekâ destekli mühendislik uygulamaları birçok avantajı da beraberinde getirmektedir.

Bu kazanımlar şu şekilde sıralanabilir;

- Detaylı analiz imkânı,
- Dünyadaki gelişmelere anlık erişim,
- Kolayca güncellenebilme imkânı,
- Doğruluğu yüksek ve hızlı sonuç alabilme imkânı,
- Sağlık çalışanların yükünü hafifletebilme,
- Çok daha az sayıda çalışan ile maliyet düşürücü katkılar,
- Sağlık sistemine erişimi az ya da kısıtlı bölgelere ulaşabilme,
- Hastalar ile doğrudan ya da dolaylı teması azaltarak hastalığın bulaşıcı riskini azaltama,
- Sürekli eğitilen modeller ile her geçen gün doğruluğu ve güvenirliği artan sistemler,
- Uzman doktorların kariyerleri boyunca görebileceklerinden daha fazla örnekle eğitilmiş modeller oluşturma fırsatı.

Belirtilen gelişmeler yadsınamaz derecede olumlu olsa da yapay zekâ destekli uygulamalar ile birlikte gelen donanım ihtiyaçları da göz ardı edilmemelidir. Modellerin eğitim süreci öğrenmesi için hazırlanan binlerce görüntüye sahip veri setleri donanım darboğazını da beraberinde getirmektedir. Şayet uygun sistemler hazırlanmazsa model öğrenimi ya çok uzun sürmekte ya da yapılamamaktadır. Doğruluk oranı bakımından oldukça yüksek sonuçlar veren pre-trained modellere bakıldığında sahip olunan parametre sayısı milyonlar seviyesindedir. Bu parametrelerin eğitim sırasında sürekli güncelleniyor olması oldukça zahmetlidir. Öte yandan birçok değerinin ne olduğu belli olmayan hiperparametre denilen değişkenler ancak sistemde denenerek bulunabilmektedir. Bu durumda uygun modeli buluncaya kadar yüzlerce deneme yapılmalı ve optimum olanın seçilmesi gerekmektedir. Sonuçta bu denli uygulamalar yaparken de hafıza, işlemci gücü, GPU desteği gibi konular teknik anlamda çözülmesi gereken kısımlar olarak karşımıza çıkmaktadır. Ayrıca pre-trained modellerde kullanılan katmanların sorgulanması da bu alanda yapılması gereken çalışmalardandır. Çünkü kapalı kutu sistemine benzetilen yapay zeka modellerinde eklenilen yapıların modelin karar vermesinde ne derece etkili olduğunun saptanması gerekmektedir.

Bu çalışmada sorgulan başlıklardan bir tanesi de sınıflandırma mekanizmasının varlığıdır. Bir çok CNN modelinde var olduğu görülen sınıflandırma katmanının gerekli olup olmadığı sorusu bu çalışmayı normal bir sınıflandırma projesinden çıkarıp modellerin optimizasyonu konusunda çıkarım yapılabilecek bir konuma getirmektedir. Büyük miktarda işlem gücü ile güncellenen bu yapıların etkili olmaması daha uzun süreli eğitime sebeb olması, daha güçlü bilgisayar gereksiminden ötürü çalışmalara negatif etki edecek durumlardır.

Bu çalışmada görüntü sınıflandırma da kullanılan CNN mimarisi üzerine uygulamalar gerçekleştirilmiştir. CNN'in içerdiği FCNN katmanı sınıflandırmanın yapıldığı katman olarak belirtilmektedir. Modelden modele yapısı ve içerdiği düğüm sayıları değişmekle birlikte çok sayıda güncellenmesi gereken parametreye sahiptir. Bu katmanda yapılacak değişikliklerin modeli hafifleştirerek donanım sorununa çözüm olabileceği düşünülmüştür. Bu fikirle birlikte FCNN katmanı tamamen kaldırılarak modelin sadece özellik çıkarım katmanı ile öğrenebilmesi ve görüntülerin sınıflandırlabilmesi denenmiş ve de Walsh fonksiyonlarının dahil edildiği bir model yapısı tasarlanmıştır. Bu sayede milyonlarca parametre gereksinimi olan modellerden yaklaşık olan 250 bin parametreli bir modele dönüştürülmüştür. Bu da VGG-19[1] ile karşılaştırıldığında altı yüz de biri anlamında gelmektedir.

Altı yüz kat daha az parametreye sahip modeller günümüzde mobil tarafta artan yapay zeka uygulamaları için de çok elverişlidir. Çünkü hem daha az işlem gücü ile sınıflandırma yapılacaktır hem de modelin daha az yer kaplamasından dolayı alan yetersizliği sorunları daha az yaşanacaktır. Öğrenme aşamasında FCNN katmanının değişkenlerinin hesaplanması için yapılacak hiperparametre çalışmalarına ihtiyaç kalmaması ile de araştırmacılar çok daha kısa sürede en optimum modeli oluşturacak yapıları kurmaları sağlanmış olacaktır. Sonrasında yapılacak düzeltmelerinde aynı şekilde daha kolay gerçekleştirilmesi sağlanmış olacaktır. Çünkü çok ağır modellerde yapılacak her bir değişiklik tüm sistemin tekrardan değişmesi anlamına gelmektedir. Yapılan çalışmanının sonuçlarına göre pre-trained modellerin yapılarının tekrardan gündeme gelmesi hedeflenmektedir.

Çalışamanın sonucunda önerilen model ile 4 farklı sınıfa ait akciğer röntgen görüntüleri alınarak modelin eğitilmesi sağlanmış ve sonuçlar analiz edilmiştir. Modelin CNN katmanı öncesinde ön hazırlık aşamasına sahip olması görüntülerin aynı boyutta ve daha temiz gelmesi açısından kritik öneme sahiptir. Ayrıca çeşitliliği arttırarak görüntülerin ezberlenme ihtimalinin azaltılması hedeflenmiştir.Eğitim ve validasyon süreçlerinde ağır modeller ile oldukça benzer başarı ve hata oranları verdiği gözlenmiştir. Tüm modeller aynı şartlara sahip test bilgisayarı ile eğitim sürecini tamamlamış ve yapılan karşılaştırmalar dış faktörlerin neden olabileceği etkilerden olabildiğince arındırılmaya çalışılmıştır. %96.15 lik test doğruluk oranı ile de karşılaştırılan 9 pre-trained modeli içersinde 4. sıraya yerleşerek ilerideki yapılması planlan çalışmalar için umut vadetmekle birlikte eğitim süresini ve de donanım ihtiyacını ciddi ölçüde azaltmıştır. Modellerin eğitim süreleri, hata miktarları, parametre sayıları, giriş-çıkış boyutları görselleştirilerek daha anlaşılır bir karşılaştırma sağlanması amaçlanmıştır. Elde edilen çıktıların analizleri yapılarak gelecek çalışmalarda dikkate alınması gereken noktalar belirtilmiştir.

1. INTRODUCTION

With technological developments, disciplines get out of their shell and interact with different fields every day. Ideas from different disciplines in many fields and innovative developments occur. In this way, problems can be solved more comprehensively. Deep learning methods, which are an important part of artificial intelligence, whose importance has increased with the support of GPUs in the last 15 years, show themselves with the solutions they offer in many challenging subjects such as image processing, voice recognition, and object detection. Thanks to the developed applications, it is aimed to train the models that can classify better than humans, with datasets that are expanding day by day, to have a much more effective structure. The unmanned solutions offered by artificial intelligence and the inclusive developments will undoubtedly prevent human errors in all fields and will enable us to get more accurate results and thus make the right decisions.

Classification of images can be done with many applications provided by artificial intelligence. The structure of the developed applications may change in line with the targets, but undoubtedly the most preferred methods are deep learning applications, with the accuracy and error rates they provide. Models developed for governments and private companies to detect people or desired objects actively exist in our lives. License plate detection software in traffic, security applications needed at borders and airports are just a few of the studies carried out in this area. In addition to image classification, many other activities involving artificial intelligence can be performed with deep learning. In this sense, deep learning is in the first place among the places that researchers look for a solution. The classification of lung x-rays, which is also the main subject of this study, is one of the sub-branches of medical imaging, and it is one of the fields that artificial intelligence is included and projects are carried out on, and the number of studies in this field is increasing day by day.

X-ray images, named after its inventor, German physicist Wilhelm Conrad Röntgen, are essentially based on the principle of penetration of X-rays by sending them to the patient and according to the amount of X-rays passing through, the tissue, organ, and bones can be detected. One of the reasons for the increase in the number of projects based on lung x-rays in recent times is due to the Covid-19 period that we are in. In fact, these days, when we realize how unprepared we are in the face of pandemics, these projects, which alleviate the heavy burden of healthcare professionals, are developed by researchers and aim to take an active role in the fight against the disease. In this way, artificial intelligence supported projects are developed to use the limited number of resources of countries more effectively and to overcome these periods with the least number of deaths. Undoubtedly, the ultimate goal of these projects is to develop models that make better classification and analysis than specialist doctors by making use of the solutions offered by artificial intelligence, and to make them accessible to researchers and healthcare professionals who have limited opportunities to fight diseases in other parts of the world.

1.1 Purpose of Thesis

X-ray devices, which are available in many hospitals of developed countries, have become a part of our lives by increasing their prevalence in the last century and X-rays, which are used to determine the diagnosis and treatment, can be used to examine many parts of the body. One of the sections examined is the lung, and many diseases such as pneumonia, opacity, cancer, chronic diseases, tuberculosis can be diagnosed through X-rays. The detection of Covid-19, which has recently been effective all over the world, can also be made by means of CXRs. Being able to detect this disease, which has caused the death of approximately 5 million people worldwide as of the beginning of 2022, as early as possible is essential to stop the deaths and the spread of the disease and the fact that diseases that affect the lungs have similar damage, unfortunately, strengthens the separation of diseases from each other. For example, diseases that cause inflammation in the lungs such as pneumonia can easily be confused with Covid-19 and can be misdiagnosed and put both patients and doctors in more troublesome situations[2].

Engineering applications are being developed to take quick action today, when it is known how important early diagnosis is. In this way, both the rate of spread will be prevented and the course of the disease can be controlled before the condition of the patients becomes critical. In addition, it is aimed to protect us from the harms of misdiagnosis and treatment with the applications to be made with the models that are constantly improving with increasing accuracy. One of the specified engineering applications is to provide the correct classification of the disease with artificial intelligence supported programs. Classification using lung x-ray images brings along engineering problems such as the need for hardware. There are many criteria to be met, such as converting thousands of images to suitable sizes, removing noise, and taking images at appropriate intervals. In addition, an image pre-processing is also required to include X-ray images of appropriate sizes in the training process. When pre-trained models with high accuracy are trained after preprocessing is completed, millions of parameters must be constantly updated. In this thesis, the main inspiration is to try to minimize the number of parameters that deep learning models have by using Walsh functions. We aim to achieve this by eliminating the fully connected neural network structure and loss function will be updated accordingly. Thus, it is aimed to prevent the problems caused by the high amount of memory required to update millions of data in each batch.

The three important contributions of this proposed model are as follows:

- Models can be trained with computers with low hardware.
- Due to the need for fewer parameters, models will be able to complete their training in less time and start the classification process.
- The number of unknowns will be reduced with less hyperparameter, thus saving time with fewer attempts.

While performing these operations, it will be among our fundamental criteria to provide success rates similar to the success rates of previous applications.

1.2 Literature Review

The subject of classification of recent lung images has been extensively studied by researchers and added to the literature. The impact of Covid-19 on the increase of these researches is undoubtedly very large. Sharing the X-ray images of the lungs they have in order to contribute to the studies carried out the health institutions enables the researchers to reach the data they want faster and to do the studies much faster. Images obtained from all over the world also allow to produce robust models. Otherwise, it is clear that the systems trained with deep learning techniques will not be healthy since the images taken from only certain groups do not reflect the generality. Although the number of classes in the literature in general is 2-3, this study tries to present a 4-class model. In this respect, this study, which aims to provide a similar success rate, will likely face different challenges.

In the study by Jain [3], there are 3 classes. These classes are Normal, Covid-19, and pneumonia. A public dataset from Kaggle is used and there are 6432 images in it. It was used 5467 of these images in training and 965 during testing. All images were converted to 128x128x3 format during the pre-processing. In order to add randomness to the data, 10 degrees of random rotation, horizontal flip and 0.4 zoom were applied. Three different pre-trained models were applied and their results were compared. These models are Inception Net V3 [4], XCeption Net [5] and ResNeXt [6]. Among these models, Xception Net [5] was determined as the model with the best performance. The success rate of the model was also measured as 97.97%.

Karakanis & Leontidis [7] conducted binary model and three-class model training in his study. The dataset contains 275 Covid-19, 275 Bacteria and 275 Normal images. Success is claimed as 98.7% in the binary model and 98.3% in the three-class model.

Ouchicha [8] named the study CVDNet. In this study, there are 3 classes. These classes are Covid-19 , Viral Pneumonia and Normal. The number of images belonging to these classes are 219 Covid-19, 1341 Normal and 1345 viral pneumonia. The images were passed through 8 Convolution (Convolution + Batch Normalization + ReLU) + Max pooling in total in 2 branches. Finally, after the 9th Convolution layer, it was combined and transmitted to the FCNN part and classification was made. Covid-19 success is claimed to be 97.2%.

Basu [9] preferred a 4-class structure in his study. Classes are divided into Normal, Pneumonia, Covid-19, and other diseases. Training was carried out using pre-trained models. Two different datasets were used and labeled A, B. Data-A has a total of 108,947 chest images belonging to 2 classes, normal and disease. In Data B, there are 1277 chest images belonging to 4 classes. The pre-trained models used were AlexNet [10] (8-layer), VGGNet [11] (16-layer) and ResNet50 [12] (50-layer). Kernel dimensions were chosen as 3x3, and the success rates were 82.98% for AlexNet [10], 90.13% for VGGNet, 85.98% for ResNet. The hyperparameters used are the SGD optimization algorithm with a learning rate 10−4 and momentum 0.9. The most successful result is 90.13% += 0.14 with 16-layer VGG, which is VGG-16.

Shankar & Perumal [13] used a pre-trained model, Inception v3 Net [4]. The model has 4 classes and these classes were chosen as Normal, Covid-19, Sars and Pneumocystis. In this study, in which Adam was chosen as the optimizer, a 94.85% success rate was measured using the method called FM-HCF-DLF.

Bhowal [14] wanted to classify Covid-19 using ensemble learning technique in this study. Unlike Weighted Arithmetic Mean (WAM), which is used in other ensemble methods, classification is made by applying Choquet Integral. The weighting schemes, which are divided into 3, are finally combined with the aggregation process of Choquet Integral. There are images belonging to 3 different classes in the dataset used (Covid-19 752, Pneumonia 1584 and Normal 1639). Images are divided into 90% train and validation 10% test. The success value was measured as 95.49%.

Hussain [15] named the study CoroDet. Models with 2, 3 and 4 classes have been developed. The model includes 2 classes (Covid-19 and Normal), 3 classes (Covid-19, Normal and Non-Covid Pneumonia), and 4 Classes (Covid-19, Normal, Non-Covid viral pneumonia, and Non-Covid bacterial pneumonia). Success rates were measured as 99.1% for 2 classes, 94.2% for 3 classes and 91.2% for 4 classes. For the 4-class model, 500 Covid-19, 400 Pneumonia-viral, 400 Pneumonia-bacteria and 800 Normal images were used for model training. A CNN model with 22 layers has been proposed. This model has 9 Conv2d layers and 9 maxpool2d layers. The last 4 layers are Flatten, Dense, Leaky ReLU and Dense layers. In addition, a batch normalization layer was added to increase the stabilization. In the 12-layer system, the success was 81% for 4 classes, while it increased to 91.2% when the layer was increased to 22.

Ibrahim [16] prepared this study based on AlexNet [10]. A 4-class model was created and it was observed that 93.42% success was achieved. The 5 layers of AlexNet [10], which has 8 layers, consist of convolutional layers. ReLU is preferred as the activation function. In the dataset, there are 371 Covid-19, 4237 Non-Covid viral pneumonia, 4078 Bacterial Pneumonia, 2882 normal images.

Sakib [17] presented a model called DL-CRC. The DL-CRC framework consists of 2 parts. These parts are DARI and CNN. They increased the number of images by creating synthetic data with data augmentation methods made in the DARI section, and models were trained with both acutal and synthetic images during the train. First, data augmentation was done with Generative adversarial Network, then the images were sent to the convolution layer and feature extraction was performed. Then the flattened outputs were sent to the dense layer for classification. There are 3 classes in total for the dataset. These classes are Normal, Pneumonia, and Covid-19. It contains 27,228 Normal, 5,794 Pneumonia, 209 Covid-19 images. Classification Success was presented as 94.61%.

Nur-a-alam [18] claimed to prevent overfitting by using the 5-fold strategy in this study. A 2-class model was created. A dataset with 1979 Covid-19 positive and 3111 Covid-19 Negative images was used. MATLAB was used as the program and VGG-19 [1] pre-trained model was used in this study. 87.34% success was achieved with HOG feature extractor. When CNN was used, the success rate was 93.64%. The proposed model combined the two to achieve an average success rate of 98.36%. It has been claimed that this success has been increased to 99.49% by adding VGG-19.

Nayak [19] compared 8 pre-trained models. These models are AlexNet [10], VGG-16 [11], GoogleNet [20], MobileNet-V2 [21], SqueezeNet [22], ResNet-34, ResNet-50 [12], and Inception-V3[4]. In this study, when the Resnet-34 model is used with Adam optimizer, it has the highest success rate with a success rate of 98.33%. Afterwards, 97.52% success was achieved by using the Adam optimizer with AlexNet. Batch size being 8, 16, or 32 did not change the success values. There are 2 classes, 203 Covid-19, 203 Normal. In the dataset, 1002 of 1550 images in total, which were formed as a result of data augmentation, were used for train, 428 were used for validation, 120 images were reserved for testing.

Wang [23] proposed the model with the name Covid-Net. The data set used has 3 classes. The classes are Covid-19, Pneumonia and Normal, respectively. The proposed model has 11.75M parameters. It is created by connecting modules called PEPX one after another. In the study, it was emphasized that it was more successful with less parameters by comparing it with VGG-19 and ResNet-50. The success rate was announced as 93.3%.

The methods used by the studies in the literature, the classes in the data set, the images included in the classes and the success rates are shared in the table below.

Table 2.1 : The summary of the literature review.

Reference	Method/Models	Classes	Number of Images	Accuracy (%)
[3]	Xception Net	Normal, Covid-19, Pneumonia	6,432	97.97
[7]	ResNet8	Covid-19, Bacteria, Normal	825	98.3
[8]	Cvdnet	Covid-19, Viral Pneumonia, Normal	2,095	97.2
[9]	VGG-16	Normal, Pneumonia, Covid-19, Other Diseases	108,947	90.13
[13]	Inception V3	Normal, Covid-19, Sars, Pneumocystis	273	94.85
[14]	Ensemble Learning	Covid-19, Pneumonia, Normal	3,975	95.49
[15]	Corodet	Covid-19, Normal, Viral Pneumonia, Bacterial Pneumonia	2,100	91.2
[16]	Alexnet	Covid-19, Viral Pneumonia, Bacterial Pneumonia, Normal Images	11,568	93.42
[17]	DL-Crc	Normal, Pneumonia, Covid-19	33,231	94.61
[18]	VGG-19	Covid-19 Positive, Covid-19 Negative	5,090	99.49
[19]	Alexnet	Covid-19, Normal	406	97.52
[23]	Covid-Net	Covid-19, Pneumonia, Normal	13,975	93.3

1.3 Hypothesis

The common demand of researchers dealing with Deep Learning models is powerful computers with high memory supported by GPUs and models that can be trained in a short time. The purpose of using powerful computers is to ensure that high-capacity datasets fit into the memory so that the model training process is not disrupted. In addition to this, another aim is to avoid spending a lot of time if the models have high parameters. Due to the large number of unpredictable hyperparameters in AI models, the training process is based on finding the best result by choosing different hyperparameters of the models. If the models contain too many parameters, this time is multiplied. Batch size, learning rate, optimizer selection, sizes and numbers of kernels in convolution, dropout rate, number of layers can be given as examples of hyperparameters. Multiple combinations of hyperparameters should be tried in order to obtain optimum values in the comprehensive research.

There are two layers in the CNN model, which is frequently preferred in image processing, these are the feature extractor and the fully connected neural network. In this study, it is hypothesized that success can be preserved since the FCNN layer is removed and necessary adjustments are made. Because the kernels in the feature extractor layer can also be updated, it is thought that model learning can be achieved. Removing thousands of variables of the fully connected layers in the FCNN layer will provide us with lighter models. In this way, models with fewer parameters can be trained in a faster time and models can be trained with a less powerful computer. This study, which includes the classification of Covid-19, is based on providing a structure that can be used by researchers with limited access to powerful computers and examining the high amount of computational overhead required in the case of training pre-trained models from scratch.As a result, this research is planned to be carried out with the hypothesis that models with millions of variables should be re-examined by removing unnecessary layers so that core models can be created.

2. CHEST X-RAY IMAGING TECHNIQUE

2.1 Chest X-Ray Imaging Technique

X-rays are a form of electromagnetic radiation with relatively short wavelengths used in medical imaging (Longer than gamma rays but shorter than visible light, infrared, microwaves and radio waves).

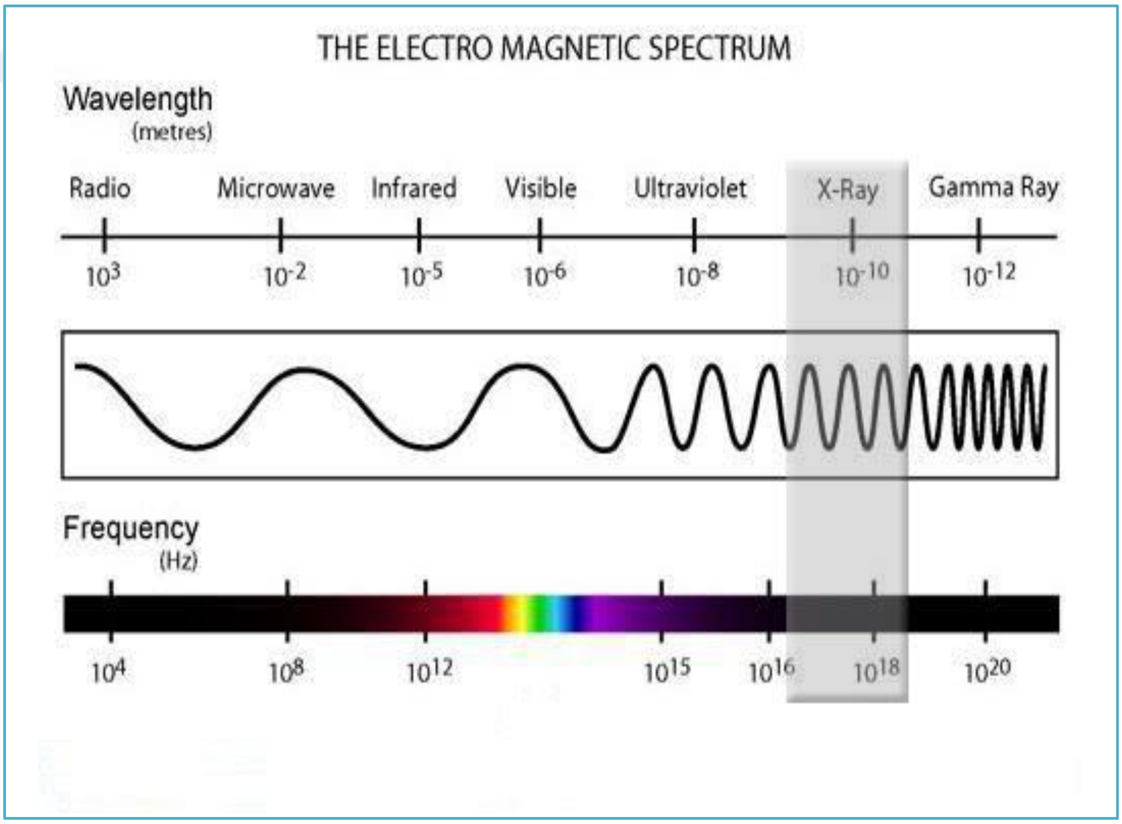


Figure 2.1 : Wavelenghts and frequencies of the electromagnetic spectrum[24].

It performs visualization according to the rate of penetration of the substances it penetrates. As can be seen in Figure 2.1, the X-ray sent collides with the material and creates an image. It is a kind of photography when generalized, but uses radiation instead of light[25].

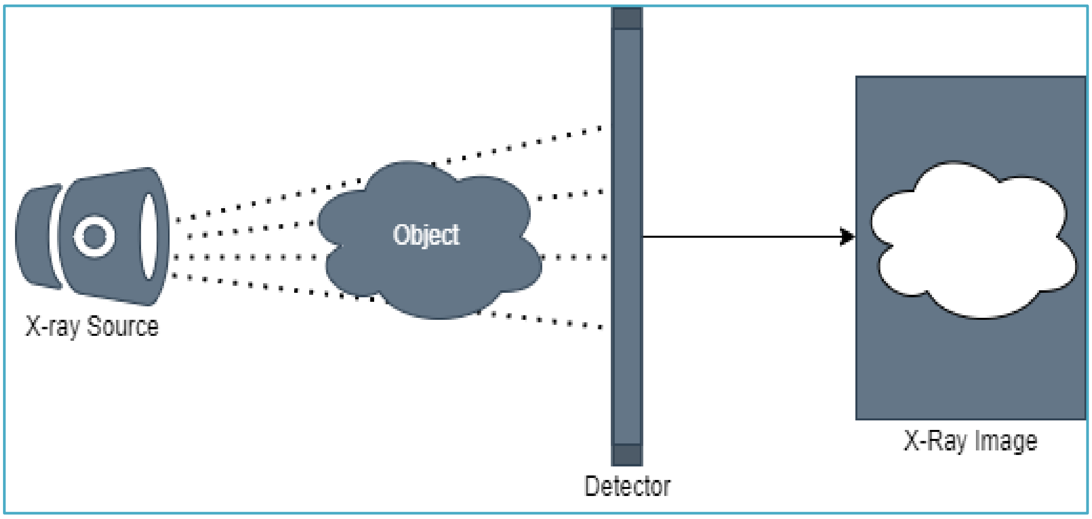


Figure 2.2 : Producing images with x-rays.

Lungs are one of the areas where X-Rays are used, which is an imaging tool to produce pictures of organs and tissues in the body. More generally, the condition of the lungs, heart, bones, chest wall, respiratory tract, spine bones in the chest region and organs and structures in injuries caused during the accident can be displayed. In addition, the diagnosis of symptomatic disease such as shortness of breath, chronic cough, chest pain can be made with chest X-rays. When it is thought that there is a problem in the heart and lungs, this method is frequently used for imaging. In this context, it is one of the most important diagnostic methods in the diagnosis of pulmonary diseases that will be examined in this study. According to the situation, anomalies can be detected by taking images of the lungs from the front and sides. In addition, chest X-ray can detect unusual fluid or air in and around the lungs [26]. These images are also used to view the chest cavity before the surgery. Thus, chest X-ray images contain solutions that are very beneficial in issues such as the size of the heart, the condition of the lungs, calcium deposits in the heart and blood vessels, and the determination of the location of the medical devices to be placed on the patient [27].

Small doses of ionizing radiation are used to give a black and white appearance to chest radiographs. The ribs and spine appear white as they absorb most of the radiation and lung tissue appears dark because it absorbs very little radiation. Although the patient is exposed to a certain amount of radiation during the procedure, it is ensured that the amount of radiation is chosen in a way that does not affect the health. Because it is fast and easy, this method is one of the methods preferred by doctors.

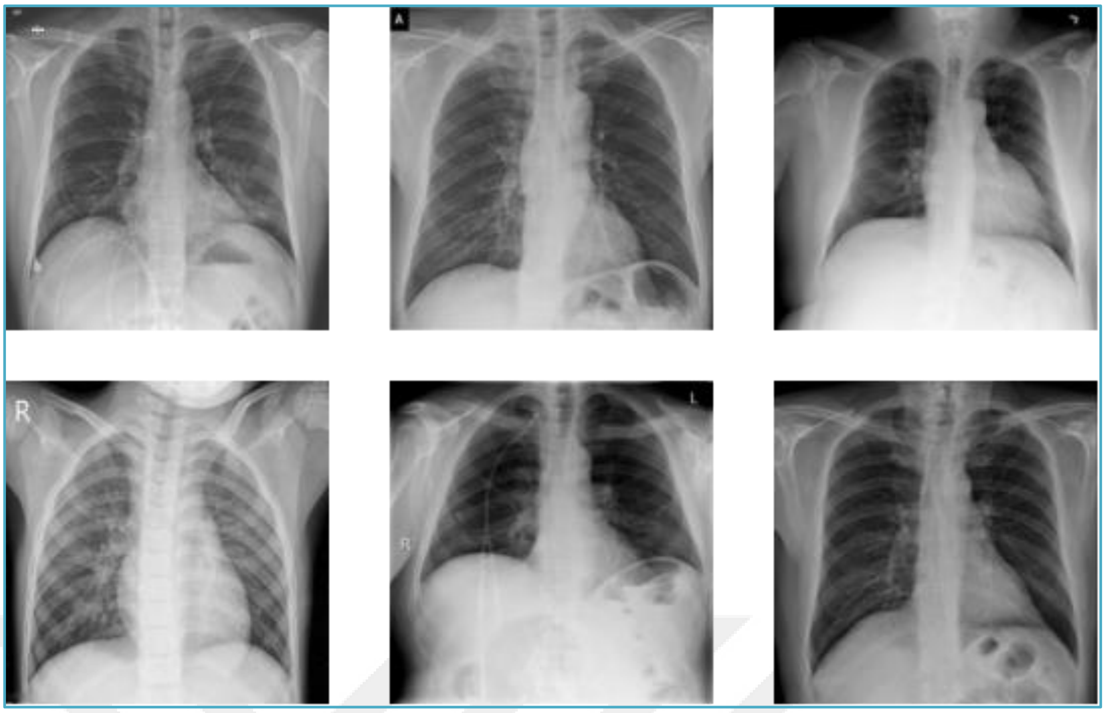


Figure 2.3 : Chest Radiographs & Chest X-Rays & CXRs.

The penetration of the CXR is dependent on the time of the patient's exposure to x-rays and the beam power. Poorly penetrated CXR appears light and soft tissues are difficult to see, especially if behind the heart. On the contrary, in very powerful rays, the CXR seems dark and it is quite difficult to see the changes in the lung. For this reason, it is important to have an optimum value [25].

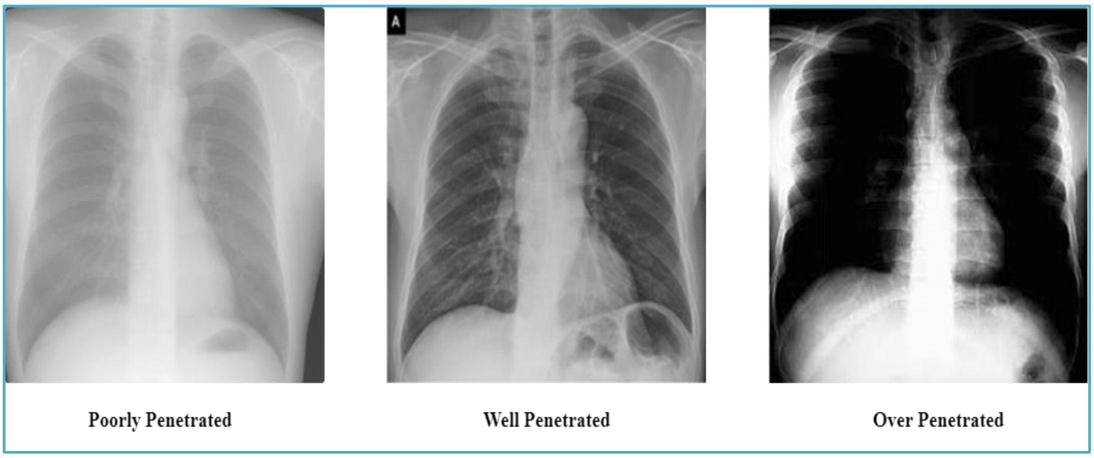


Figure 2.4 : The view of poorly, well and over penetrated Chest X-Ray Images[28].

The diagnosis of diseases affecting the lungs such as Covid-19, pneumonia, which we have heard frequently recently, can also be detected by x-rays.

Taking Pictures of X-rays:

Imaging takes place in a special room with an X-ray camera attached to a large metal arm. The X-ray machine consists of a wall-mounted box-like apparatus and a special plate that records the images obtained. The X-ray generating tube is positioned approximately 6 feet, or 180 cm, from the area where the plate is located. The patient is also positioned between the x-ray tube and the plate.

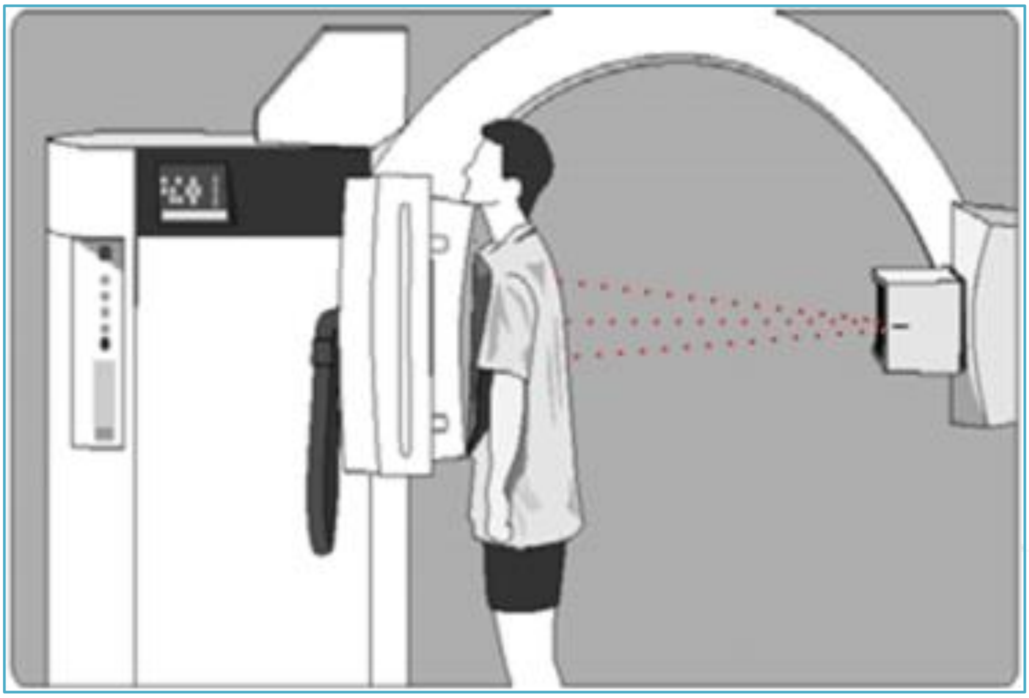


Figure 2.5 : Positioning of X-Ray machine and patient[29].

X-rays sent to the patient show different behaviors in tissues and organs with different densities. It is reflected back or penetrating according to the density. In order to catch the rays coming out of the body, a film plate is placed on the front and imaging is performed with X-rays passing through the patient. In the graph, the 3-dimensional structures are displayed in two dimensions, and the resulting x-ray images are evaluated by radiologists.

X-ray filming takes place very quickly and is not painful for the patient. But during this time, the patient is exposed to radiation. For this reason, as few X-ray films as possible should be taken in children, adolescents and pregnant women. Imaging takes a total of 15 minutes, including preparations etc.

Types of Chest X-Rays:

X-ray methods vary according to the patient's condition. In general, the patient is positioned standing, but for patients who have difficulty standing, it is possible to take the image in the horizontal position.

Types of X-Rays can be listed :

- Posteroanterior (PA) Chest Radiograph: It is the most commonly used method. The patient stands upright and leans his chest against the metal plate of the X-ray machine. The hands and arms are held at the sides, the shoulders are pressed against the plate, the chin is placed on the upper line of the plate. After these procedures are completed, the patient is asked to hold their breath. X-rays are sent to the patient to create an image of the front of the chest. The images used in this thesis are the images obtained by this method.
- Lateral Chest Radiograph: The standing patient leans sideways against the metal plate of the x-ray machine. Raising his arms up, he holds his breath until the filming is over. In this way, an image of the side of the chest is formed.
- Anteroposterior(AP) Chest Radiograph: The patient lies in the supine position with the rib cage against the x-ray plate.
- Lateral Decubitus Chest Radiograph: With this method, where the patient lies sideways on the table with the x-ray plate, the areas of fluid collection in the lungs can be determined.

There are many situations that can affect the image during capturing. In some cases, contrary to popular belief, it does not affect the images. The rules and general conditions to be considered during shooting can be listed as follows:

- Being hungry or full does not affect the x-ray image.
- Clothes above the waist should be removed and a hospital gown should be worn.
- Jewelry, glasses, piercings or other metal-containing objects should be removed.
- Doctors should be informed if there is a heart valve, pacemaker or in-body devices implemented in that area.

- X-rays should not be taken for women during pregnancy unless necessary.

X-Rays Interpretation Process:

A chest x-ray produces a black and white image. Colors and shades in the x-ray are critical for the diagnosis to be made. The meaning of the colors will also be examined in detail in Section 3.4. Structures that block radiation appear white, and structures that transmit radiation appear black. The bones are white in color because they are dense. The lungs are filled with air, and since the radiation can travel through the air, it passes easily, and the lungs have a gray and black appearance.

In terms of evaluation criteria, the dose of X-ray given is very important, on the other hand, bronchi and possible masses have an importance that will affect the diagnosis in the evaluation. In terms of the condition of the lungs, the healthy lung x-ray should be completely black. On an X-ray, the bones appear white, but the lungs appear black. The absence of any spots or smoky appearance on the transparent black indicates a healthy lung. Shortness of breath due to smoking is reflected on the lung film as white smoky spots and the lung film is blurred, which means the bronchi are full. This is also a sign of lung cancer. A clean lung film is smokeless and clear.

Signs of pneumonia or respiratory tract infections are even more pronounced. A patient with pneumonia may have white dots that will be clearly visible on the chest x-ray. These points are called silhouette signs. In pneumonia patients, the boundaries that should be in imaging are lost. For example, in a healthy person, the borders of the heart are clearly marked because the blood in the heart has a higher density than the air spaces in the lungs, which affects the X-ray image, but in the patient exposed to pneumonia, the borders between the organs and structures disappear[25]. Situations similar to pneumonia may occur in Covid-19. Usually produces ground glass and consolidative opacities with a bilateral, peripheral, and lower lung distribution. Changes that occur can be detected with chest x-ray images as well as with CT [30]. In fact, in a study conducted in this area, it was concluded that CT scans are more sensitive in detecting Covid-19 [31]. However, it should be considered that CT images contain more radiation and may not be suitable for younger patients.

Another diagnostic method is Lung ultrasound. In this method, which detects objects according to the spread of sound, disease diagnosis can also be made. In one study, the performances of chest x-ray in detecting pneumonia were analyzed and it was observed that ultrasound was more effective in the study. The sensitivity was 0.95 in ultrasound and 0.6 in radiography [32].

In this study, a model was not created with images containing CT or lung ultrasound due to the presence of X-ray images, but in the future, if hospitals and institutions share datasets consisting of images taken by different techniques, this study can also be tried on the specified images and the success rates are compared therefore the most suitable imaging method for deep learning can be revealed.

2.2 Detection of Covid-19 in Chest X-Ray Images

Covid-19 has been in our lives for more than 2 years since 2019. According to the latest data announced by the World Health Organization (as of December 28, 2021), more than 280 million diagnoses have been made worldwide, and Covid-19 has caused 5.4 million deaths and 8.6 billion vaccines have been applied as of December 23, 2021 [33]. It seems that the mutations described as the new variant of concern (VOC) are at a level that can affect the number of deaths [34]. Unfortunately, this situation also causes concerns as the vaccines developed do not work and may negatively affect the natural immune system [35]. The biggest threat is being in heavy traffic between dense communities likely to be affected by infectious diseases such as Covid-19. The less distance people get from each other, the more likely the disease will spread.

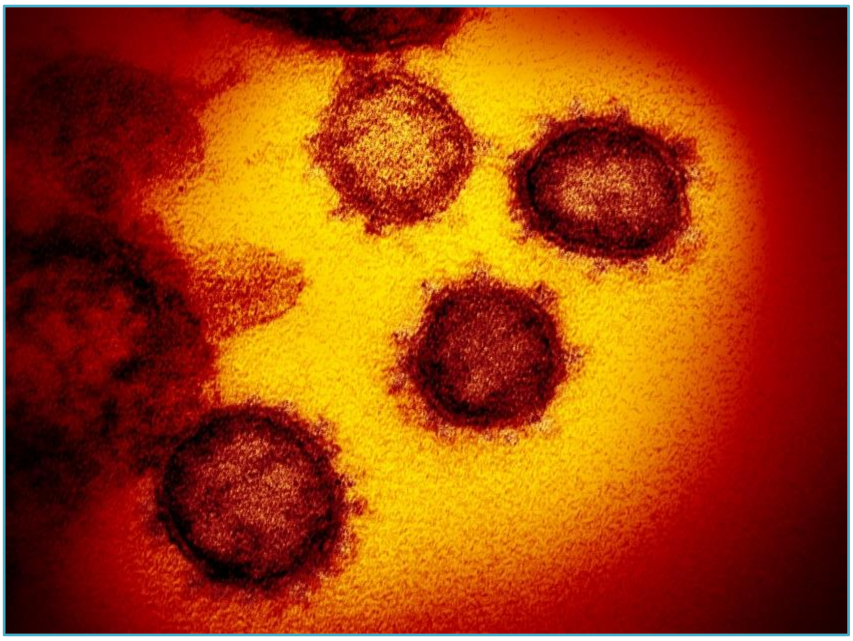


Figure 2.6 : Image of the SARS-CoV-2 (2019-nCoV) under the electron microscope[36].

When viewed proportionally, it can be said that mortality rates are higher in developing countries in the light of the data received from the World Health Organization. In the most recent data (as of 28 December 2021), the top 5 countries in the number of deaths per 100,000 in the last 7 days are listed as Trinidad and Tobago, Georgia, Hungary, Croatia, Poland, respectively.

Insufficient financial resources and inaccessibility to health services can be listed as one of the main reasons that negatively affect this situation. Stubbs [37] states that an immediate need of $2.5 trillion is required. However, the amounts provided are only around 10% as of July 31, 2020.Financial issues like these undoubtedly make us vulnerable to new variants as well as severe fatalities.

Covid-19 was first discovered in the 1960s [38]. According to various studies, Covid-19, called SARS-CoV-2, is 96% identical at the whole-genome level to a bat coronavirus [39]. The incubation period is estimated to be between 7 and 14 days [40]. Although the symptoms shown are very diverse, the most frequently reported ones are fever 83%, cough 82%, and shortness of breath 31% [41]. In children, this condition is generally asymtopic or milder than in adults [42].

Coronavirus, a single-stranded large RNA virus, has the power to infect both humans and animals of various species. Although symptoms are not clearly observed in asymptomatic patients, effects such as fever, cough, nasal congestion, fatigue and other signs of upper respiratory tract infections can be seen in less than 1 week in symptomatic patients. Also, according to Velavan & Meyer [43], pneumonia usually occurs in the 2nd and 3rd weeks of symptomatic infections and causes decreased oxygen saturation, blood gas deviations and these changes can also be seen with a chest X-ray. This infection brings with it many severe symptoms seen in pneumonia. These symptoms are briefly dyspnoea and severe chest sysmtons seen in 75% of patients and can be visualized by computed tomography [44].

Many mental problems, which are thought to be caused by the effect of the pandemic, are widely observed in humans. Most worryingly, post-traumatic stress disorder (PTSD) is more common in people. Another issue affected by Covid-19 is the burden on healthcare workers who are exposed to long working hours to treat the disease. Since the disease has lasted for more than 2 years, many emotional illnesses have occurred. These are stress, depression, irritability, insomnia, fear, confusion, anger, frustration, boredom, and stigma associated with quarantine [45].

According to C. Wang [46], there are many clear evidences for the proliferation of these mental illnesses during the Covid-19 pandemic. According to a study conducted in China in January and February 2020, 54% of the participants experienced psychological problems during the Covid-19 pandemic. It was stated that effects of Covid-19 was seen in participants as moderate to severe anxiety and depressive symptoms.

Covid-19 is not in a disease group that has a cure at the moment. It is aimed to overcome this disease by increasing the immunity of the patient with various vaccine studies and supplementary foods. On the other hand, essential conditions such as transportation, production and education also cause the spread of the disease, leading to the lack of adequate resources for treatment. To give an example, the decision of many governments to continue education face-to-face requires that both students and teachers make various arrangements [47]. In addition, the fact that schools and universities are open and the young population, which is seen as these asymptomic patients, has an increasing effect in the epidemic, causing health institutions to take a heavier burden. According to Delahoy [48], "B.1.617.2 (Delta) variant of SARS-CoV-2" with high permeability is increasing due to the return of many prekindergarden children to school and it is becoming increasingly important to monitor the status of the disease in groups under certain age.

As mentioned above, there are many methods that will increase the immune system and follow the course of the disease. These include bed rest, supportive treatment to maintain energy supply, maintaining water and electrolyte balance and monitoring vital signs and oxygen saturations. In cases where saturation is low, oxygen mask and therapy should be applied. As stated in the paper, hydrogen-oxygen inhalation is also among the applications that can be tried [42]. It is important for patients with mild symptoms or asymptomatic patients who do not show any symptoms to stay at home and be constantly monitored. In this regard, artificial intelligence supported projects will be very useful. In this way, the number of patients coming to the hospital will be reduced, and appropriate resources will be provided for patients with more severe illness.

In order to prevent infectious diseases such as Covid-19, hand and respiratory hygiene specified as standard by the world health organization should be paid attention and necessary equipment should be used [49]

Currently, many studies aim to produce a suitable and effective vaccine. International organizations and governments are in the race for vaccines and they aim to produce vaccines that can be effective against the latest variants. Based on the data received as of the last week of December, there are 31 vaccine approved, 548 vaccine studies in total [50].

As engineers, we are participating in this race with AI systems, and we are trying to come up with products that can help prevent this disease effectively and cost-effectively in this fight as soon as possible. The main purpose of using AI is to do the work that requires a lot of manpower by software. In addition, it is to achieve better results by increasing the speed and accuracy in applications such as data analysis and laboratory result predictions, thus providing more support to healthcare professionals. In this way, decision-making and patient management will be maintained in a healthier way [51].

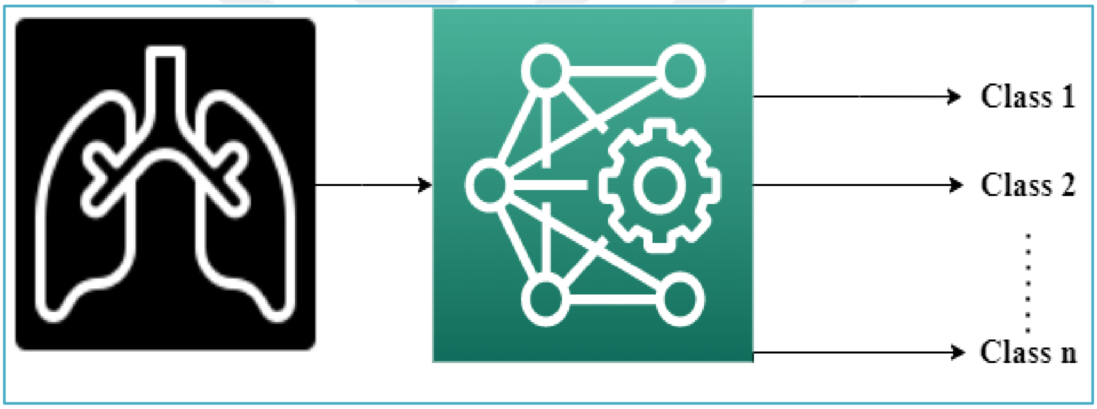


Figure 2.7 : Deep learning and CXR.

The processing and interpretation of chest radiographs, which are a part of this study, can be performed in a much faster time with AI support. According to the information given by McCall [52], while the manual reading takes 15 minutes, this time can be reduced to seconds with AI, and the diagnosis of Covid-19 can be detected much faster thanks to the information obtained from the images.

3. CLASSIFICATION OF CHEST X-RAY IMAGES BY DEEP LEARNING METHODS

In this thesis study, classification of lung images is made using deep learning methods. Since the models give more successful results in the training process and do not cause overfitting, the preprocessing is one of the steps that are useful, although not mandatory, at the beginning of deep learning models. In addition, since the images have different sizes, it can be considered as a part of the preprocessing in converting them to images of the same size. In this section, it is explained in detail what the CNN architecture is and what it uses, together with the preprocessing steps. Afterwards, the innovation made in this field with divergence-based convolutional networks is also included. Finally, the stages seen in the classification process of lung images with these methods are outlined in general terms. The parts where the possible differences in the data set are corrected and the weights of the details required for classification are increased in the model, thus aiming to increase the success rate in the model are explained in this section.

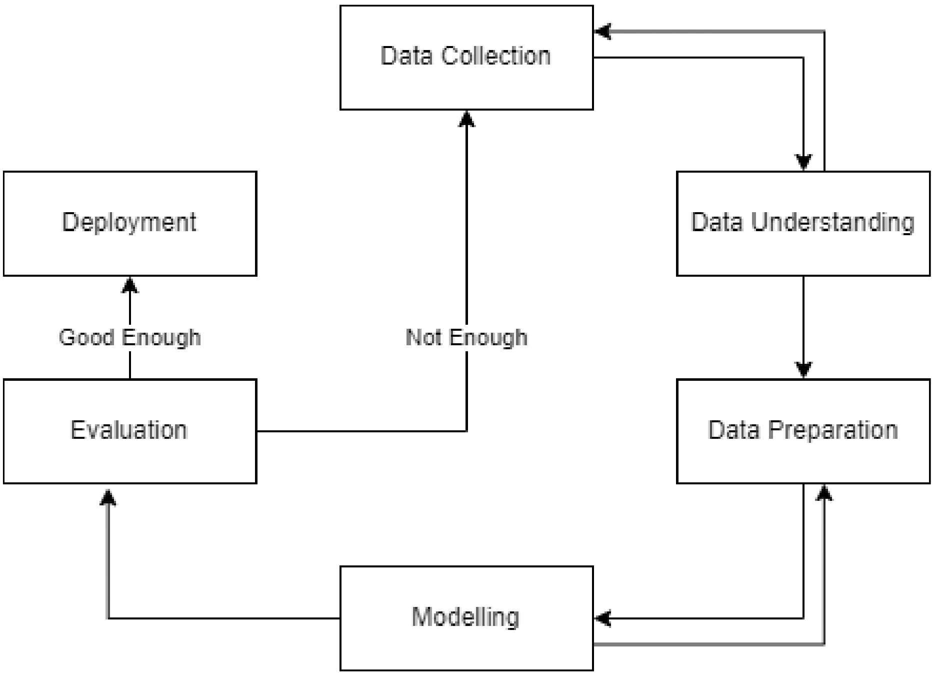


Figure 3.1 : Diagram of the model creation.

3.1 Data Preprocessing

As it can be seen in Figure 3.1, after the collected data is interpreted, the preparation period begins. According to most researchers, the most comprehensive and long-lasting stage in studies with data is the preparation stage. When it comes to data, it is likely that the clusters formed by the rows and columns in the tables come to mind but datasets containing images and videos are also involved in this demanding preprocessing part and this section will describe the most common preprocessing techniques in image classification.

Data Cleaning:

In the case of obtaining public resources of the images taken, the first thing to do is to determine the data that does not actually belong to that class. Thus, the probability of making wrong interpretations is reduced by being affected by the wrong data. Many models operate on the assumption that the data from the dataset is clean, of good quality, and noise-free, but in reality, such a dataset is unfortunately not possible.

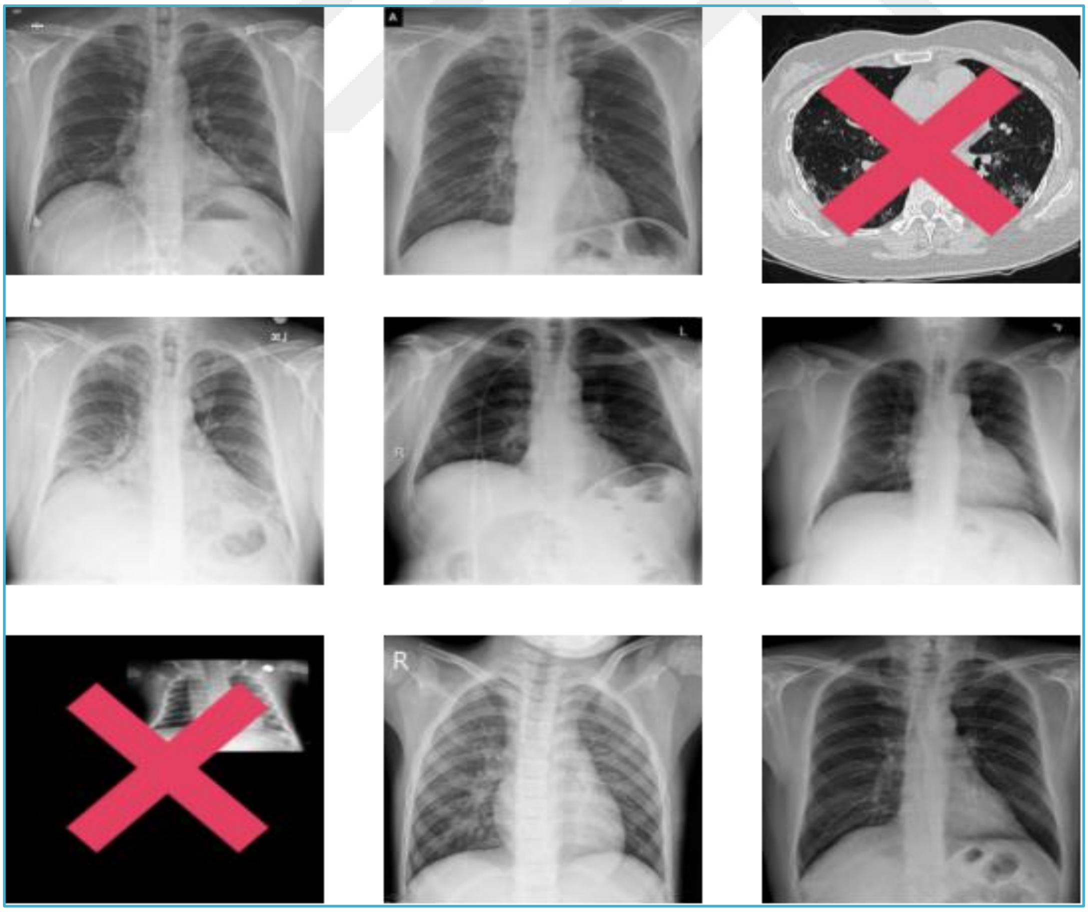


Figure 3.2 : Chest X-Ray And CT Scan images in the data cleaning phase.

As indicated in Figure 3.2, images that should not be in the dataset need to be cleaned with various software or manually. Otherwise, the model will have difficulty in having a robust learning graph.

Normalization:

The images received may be images with different intervals. In general, each pixel in the image consists of pixels with values in the 8-bit range (in the range of 0 to 255). However, this range may vary depending on the sensor characteristics of the cameras. What is usually done is to take the image between 0 and 1 or between -1 and 1. In this way, we aim to prevent that the numbers become too large and take up too much space in the memory in the operations to be performed on these pixels, and we try the images to reduce to a certain range so that all images contain values in the same plane.

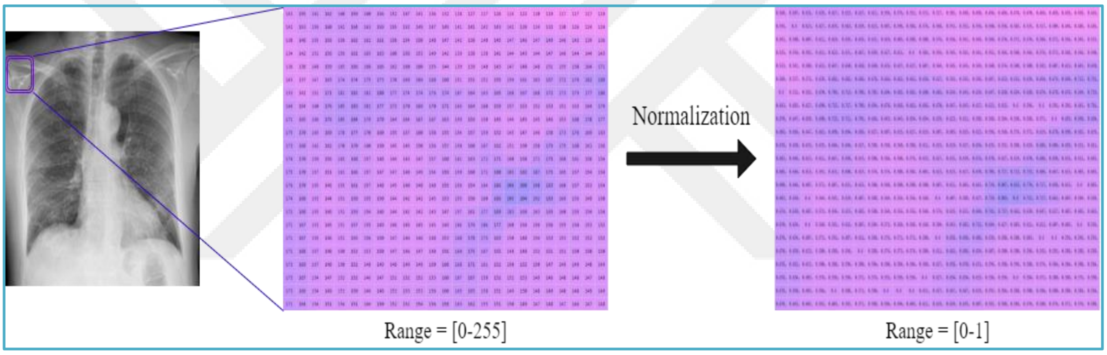


Figure 3.3 : Image normalization from 0-255 to 0-1.

Linear Scaling: With the specified operation, the numbers in the image will be in the range of 0 to 1. The values of the image between 0-255 in Figure 3.3 are reduced to between 0 and 1 by scaling.

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.1}$$

However, if the dataset is not uniform and there are very different outliers, this will cause the normal pixels in the image to be lost. Z-Score is used to prevent this situation

Z-Score: Z-score is a more advanced method than linear scaling and makes mean(u) = 0 and std = 1. In this way, it helps to solve the problems that may occur when the outliers are very different.

$$x' = \frac{x - u}{\sigma} \tag{3.2}$$

Data Transformation:

Transformations of the images taken are very critical because it is very possible for float values to occur in the images as a result of the operations to be performed. In this context, if images are stored in variables with integer values, many features will be lost. In addition, models developed by libraries such as Tensorflow or PyTorch have variable types requested to start the training process. In another case, processing may be required by transferring the images to the frequency plane. In here, transformations like fourrier may become necessary for images. Another example is about the dimensions of the images taken. If RGB images are desired as grayscale, as one-dimensional, it is necessary to reduce them from 3 dimensions to one dimension.

In the example given in Figure 3.4, the desired image type for the model should be .jpg and grayscale, and all received images of different types and sizes have been converted. In this way, it has been tried to avoid any errors when all images are used by the program.

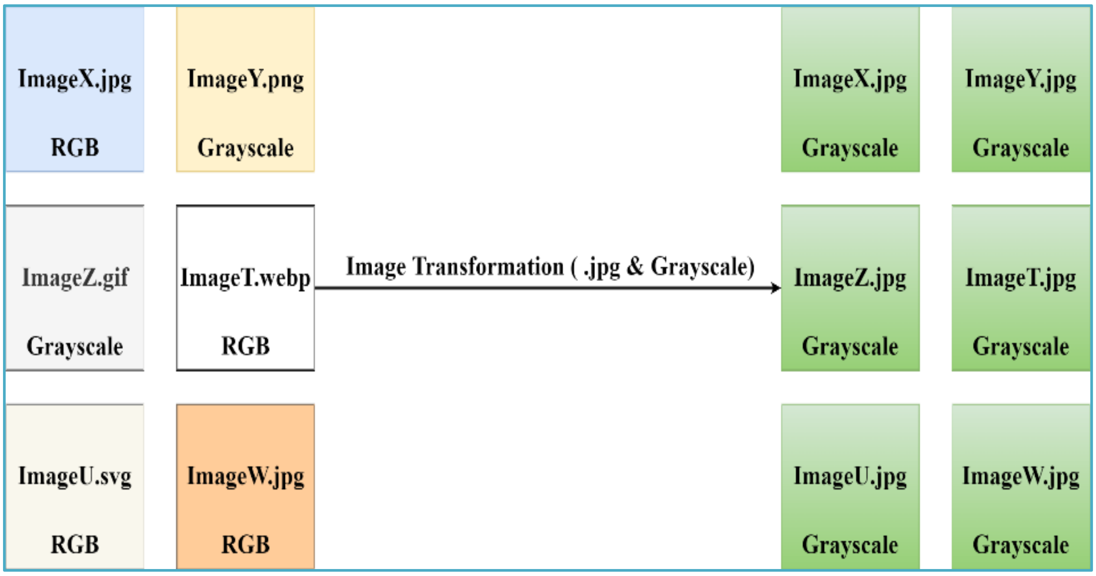
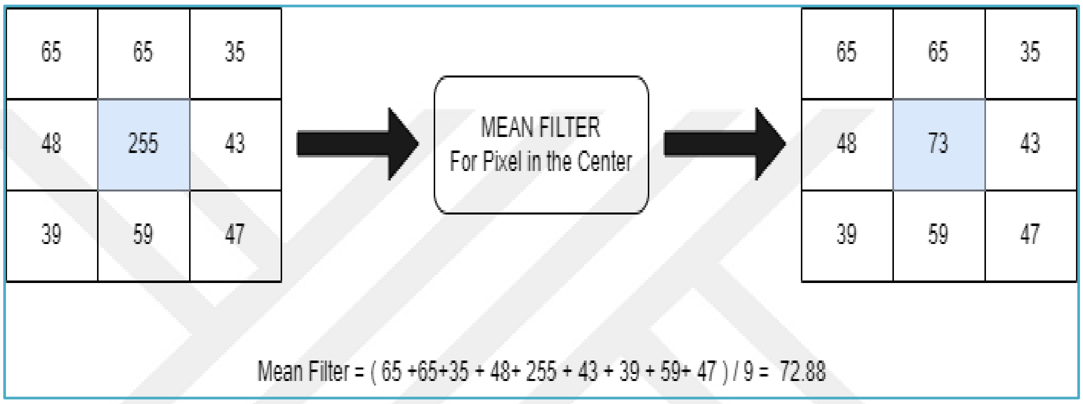


Figure 3.4 : Image transformation process.

In this study, it is aimed to prevent the processing load by converting the images taken to gray format. For similar reasons, images usually need to be transformed in the preprocessing part. Most of the time, this process is done with normalization.

Correction of Sensor Based Errors:

These errors are the errors caused by pixels with significant high or low values relative to their surroundings. Generally, the correction process is performed by taking the average of the neighbor pixels.



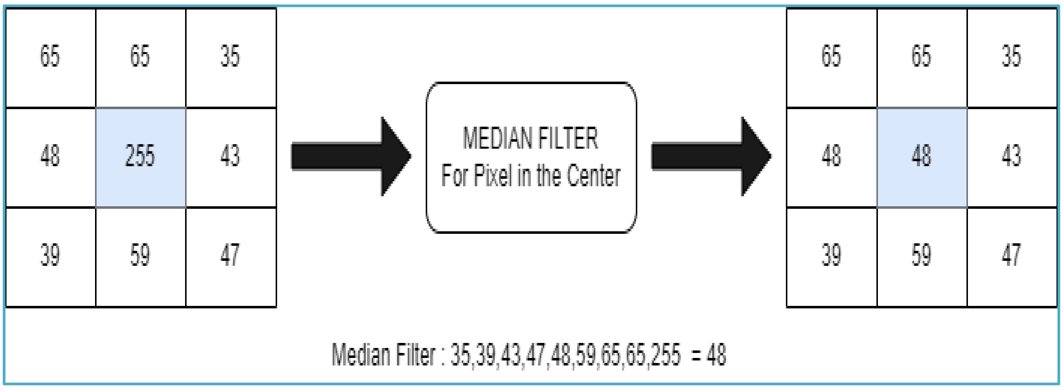
65	65	35
48	255	43
39	59	47

65	65	35
48	73	43
39	59	47

Mean Filter = (65 +65+35 + 48+ 255 + 43 + 39 + 59+ 47) / 9 = 72.88

Figure 3.5 : Mean filter example.

In the mean filter example given in Figure 3.5, the average of all values in the neighboring pixels was taken and the value of the central pixel was found. In this way, a value that converges to neighboring pixels is obtained.



65	65	35
48	255	43
39	59	47

65	65	35
48	48	43
39	59	47

Median Filter : 35,39,43,47,48,59,65,65,255 = 48

Figure 3.6 : Median filter example.

In the example in Figure 3.6, the median filter is explained and the desired pixel value and the neighboring pixels are ordered from smallest to largest and the middle value is taken. The mean and median filters, which are also shown in Figures 3.5 and 3.6, are often preferred in solving sensor-related problems.

Histogram Equalization:

In cases where the contrasts are very close to each other, the loss of the properties and properties of the objects occurs. This situation is tried to be corrected with various histhoram equalization methods. In histogram equalization, by spreading all the color ranges collected on one side to the general, it is ensured that the objects in the image appear more clearly. There are many histogram equalization methods in the literature, but in this study, sharper images were obtained by ensuring that the regionally equalized areas with the CLAHE method do not cause loss of meaning.

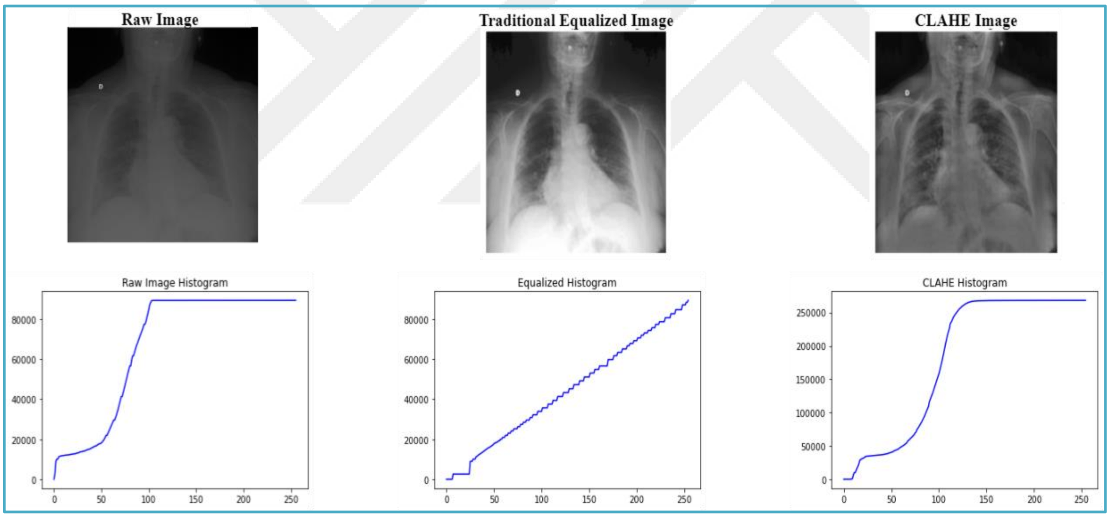


Figure 3.7 : Comparison of the histogram equalization methods and their outputs.

As can be seen in Figure 3.7, there are almost no pixels with a pixel value above 100 in the raw image in the dataset, meaning that the picture consists of pixels collected on the dark side. In the Traditional Equalization, it is ensured that the same number of pixels are formed in all intervals, but in this case, it is considered dangerous in terms of losing the characteristics of the image. In addition, noise can increase when this method is applied. The CLAHE method, which has been used frequently in recent times, divides the image into regions and makes contrast enhancement regionally.

In this way, it contributes to the better learning of the model by sharpening the details at the same time without distorting the image.

Filter Usage:

The use of filters is part of the convolution operation, which will be explored further in the next section. The new images formed as a result of moving smaller sized matrices, also called kernel or filter, on the image will be different from the old ones. This change in the image changes according to the values contained in the matrix used. Usually the goal is to highlight or suppress some object. A low-pass filter is used to highlight large, similarly-toned areas and reduce very small details. High-pass filters, on the other hand, are used to sharpen small details and bring out as much detail as possible. Edge detection filters are used to identify linear structures such as paths and field boundaries.

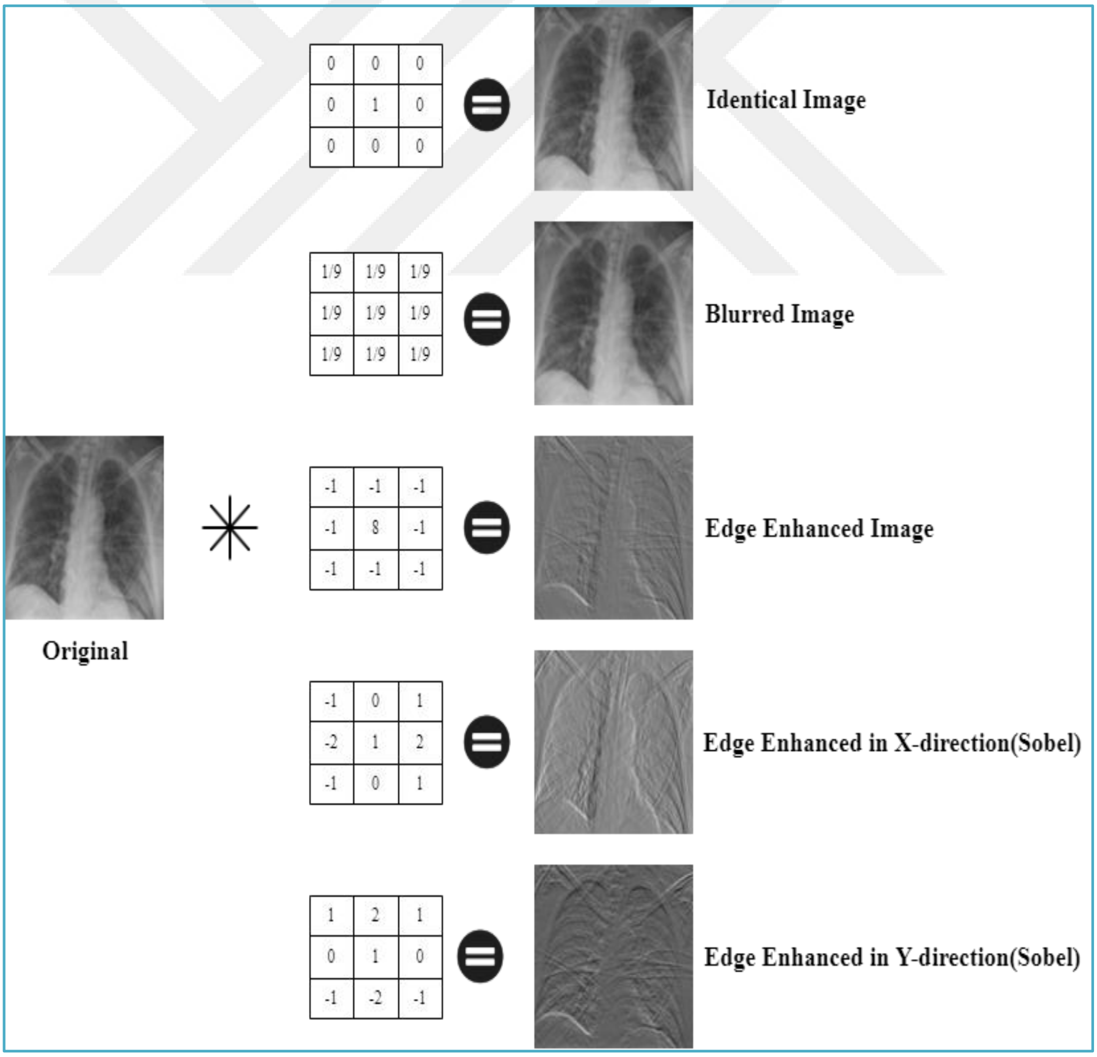


Figure 3.8 : 5 Different filters and their convolution operation results.

In the Figure 3.8, the images resulting from the convolution of the original image with 5 different filters are shared. As can be seen, we can reveal the details on the axis we want in the image, or, depending on the purpose, we can prevent pixel distortions such as salt and pepper noise.

Image Rotation:

There is a widespread belief that adding randomness to the images increases the success of the model. In this way, the model learns the image instead of memorizing it. Rotating or flipped images at random degrees can be given as an example of this situation. In addition to this situation, adding a certain amount of noise to images is a procedure tried in the literature. When the number of images needs to be increased, applications such as rotation, flipping, adding noise, changing concentration provide the basis of the process.Image Rotation can be listed as one of the Data Augmentation Tecnhiques.

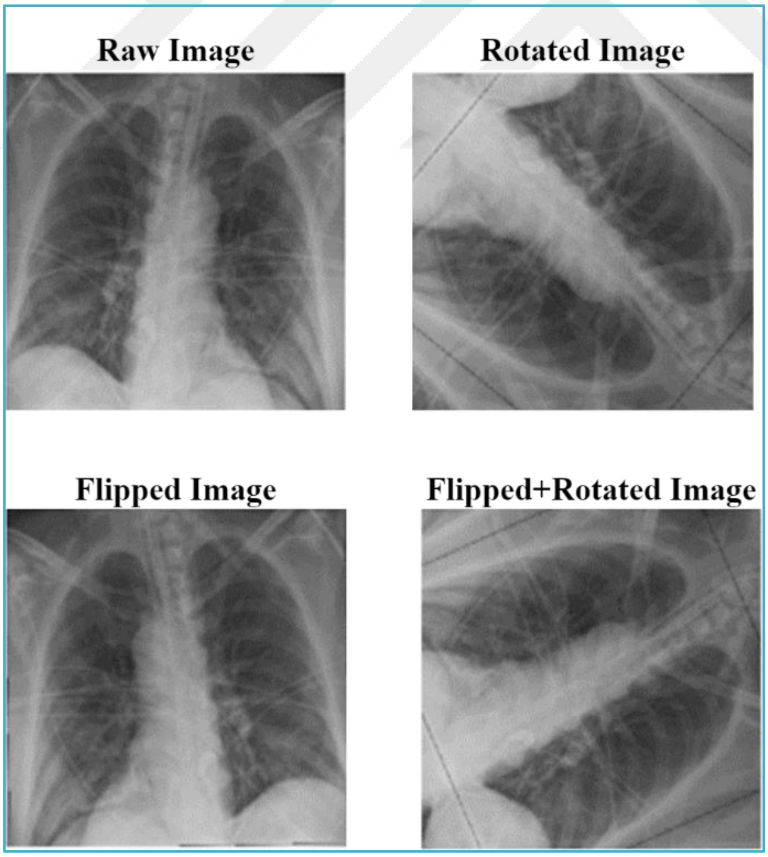


Figure 3.9 : Adding randomness and augmentation with rotating the images.

Reducing the Image Size:

We have converted high resolution images to low resolution images to lighten the load on the computer. The images we have have a very high pixel count due to their size. Considering that the average dimensions of the obtained images are around 1200x1200x3, it is necessary to process 1440000x3 on each image. We have achieved a very high amount of processing savings by making the dimensions of all x-ray scans 128x128x1 without losing the meaning of the image. The original look would probably have given better results, but in terms of cost it's very likely to overwhelm low-power computers.

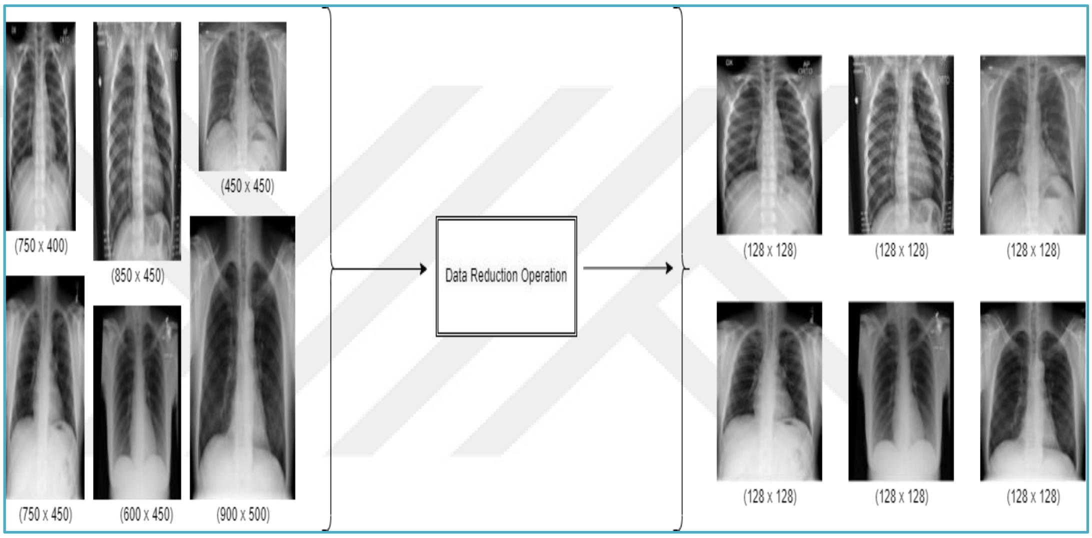


Figure 3.10 : The view of data reduction in images size.

Data Augmentation:

If the available data is too small, the probability of overfitting the model increases. Datasets with more images are created by duplicating the existing data with augmentation methods. In this way, it not only prevents the memorization of the image, but also reproduces the data in a way, resulting in healthier and more successful models.

There are many ways to augment the data as follows:

- Flipping
- Rotating
- Scaling

- Cropping
- Adding Noise

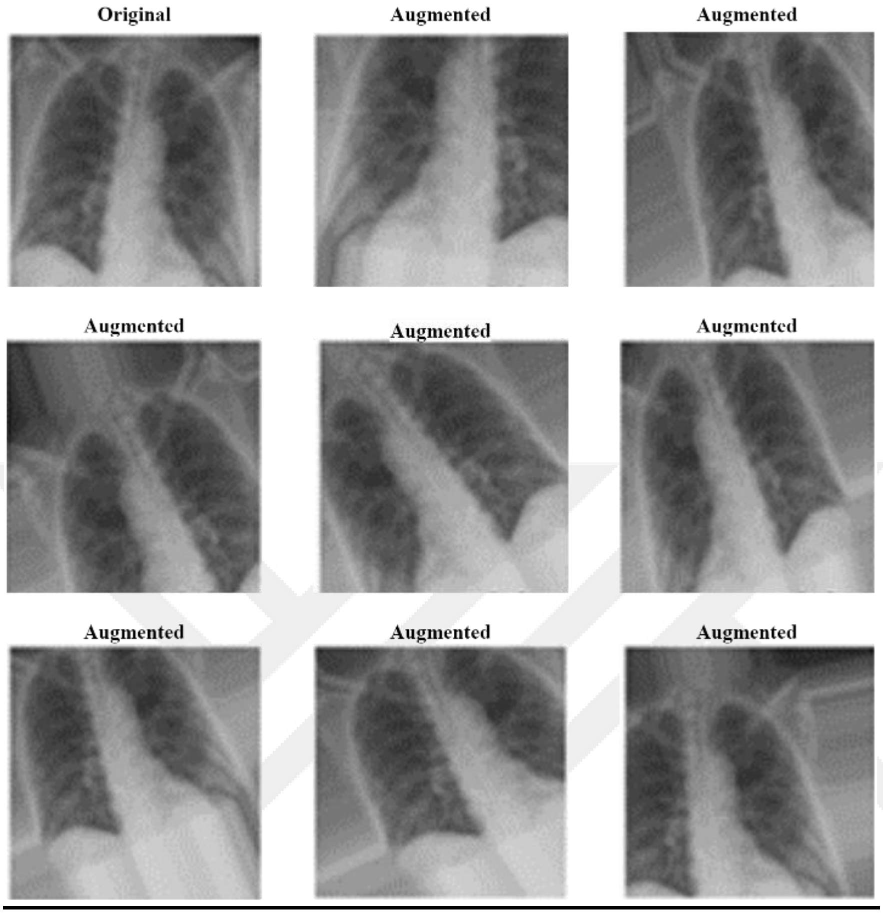


Figure 3.11 : The view of data augmentation.

In this study, it was tried to close the 7-fold difference between the classes with image reproduction techniques. The Image Datagenerator in the Tensorflow library was used.

3.2 Deep Neural Networks

CNN is one of the deep learning methods often used for the classification of images. It is possible to divide the CNN layer into 2 parts. These layers are Feature Extraction and Fully Connected Neural Network. While convolution operations are performed in the Feature Extraction layer, operations of neurons connected to each other are performed in FCNN. The images that go through the preprocessing process are first sent to the feature extractor layer, then the values in the matrix form are sent to the flatten layer to be displayed in a single column and are made ready for FCNN.

Afterwards, the learning process continues in the fully connected layers and when it comes to the output layer containing as many neurons as the number of classes, the model makes an estimation of which class the image belongs to. The neurons with the greatest value represent the classes predicted by the model. Parameters are updated with the back-propagation method, depending on whether the estimation is correct or incorrect. These stages will be analyzed in detail in this section.

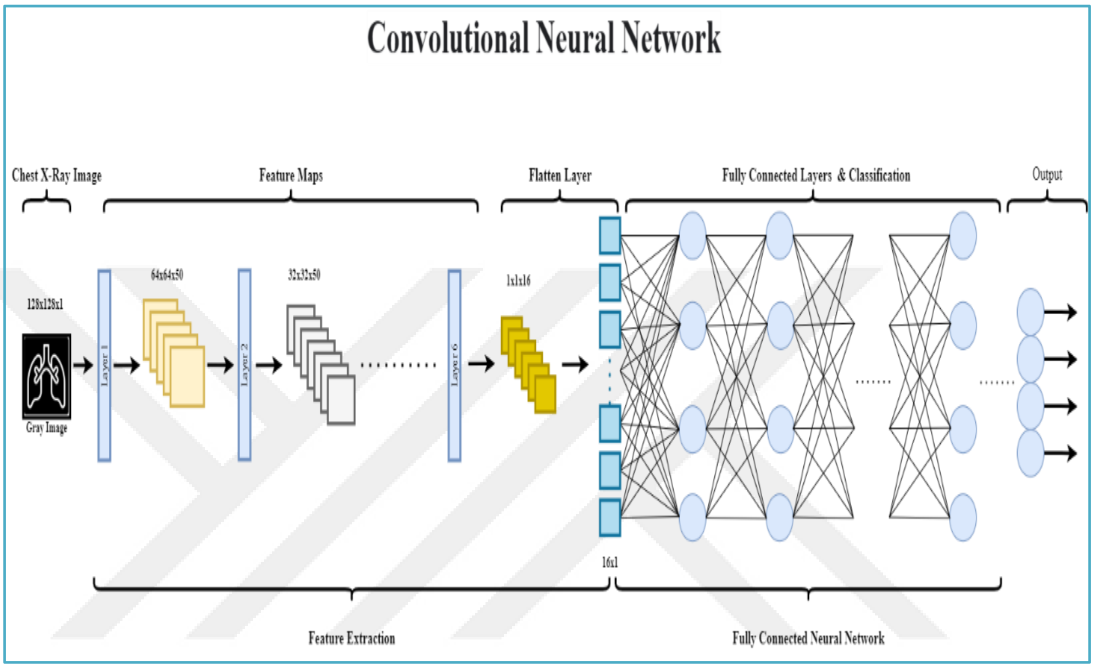


Figure 3.12 : Traditional Convolutional Neural Network structure.

Feature Extraction Layer:

It is located between the pre-processing part and the flatten layer. With the activation functions, nonlinearity is ensured and the pooling layer is added to prevent the data load from increasing, and the size of the matrix formed by taking the most important regional pixels is reduced.

The points to be considered in this layer and the processes can be listed as follows:

Input-Output Sizes:

Both the sizes and dimensions of the images entering the model change with various operations in the feature extraction layer. When the CNN model is created, the dimensions of the filters in the layers should be selected by considering the changes in sizes of feature maps between the layers. To give an example of these changes, the sizes of the output change depending on the stride and padding values during the convolution process. In addition, since the feature maps get smaller with pooling operations, the output size changes in this part. It will be explained in detail how the changes occur in the convolution, stride, pooling and padding sections.

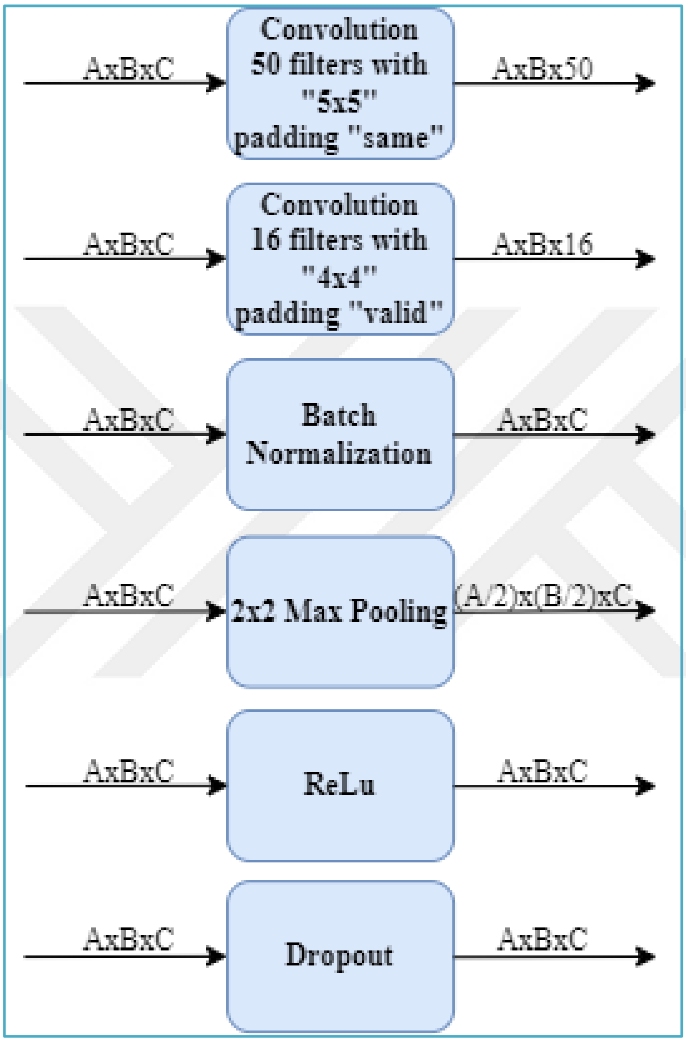


Figure 3.13 : Image size chance after operations (convolution , pooling, ReLu, normalization etc) in CNN.

In Figure 3.13, the processes applied to the images during the feature extraction layer are listed. In addition, it is also shown how the sizes of the images before the operations will change after the operation. As can be seen, while batch normalization, ReLU and dropout do not cause a change, the size has decreased in the pooling layer, and the size of the image changes depending on the number of filters in the convolution layer.

Convolution Operation:

Convolution, one of the most important subjects of signal and image processing, is a mathematical operation and is frequently used in applications with different dimensions such as speech processing(1D), image processing(2D) and video processing(3D). Although the purpose of use varies from application to application, it is preferred for targeted improvement in the image. For example, blurring, sharpening, embossing, edge detection, noise reduction can be done with convolution operations. As mentioned in Section 3.1.6, a smaller matrix, called kernel, than the normal image is shifted on the image by masking, and as a result, the process of revealing the new image is called convolution. The filters used during the process in CNN are called kernels.

$$g(x,y) = w * f(x,y) = \sum_{dx=-a}^{q} \sum_{dy=-b}^{b} w(dx,dy) f(x+dx, y+dy) \tag{3.3}$$

The matrices that emerge after the convolution operation, whose equation is given above, are also called feature maps. The critical point is that the dimensions of both matrices must be the same. In addition, the defined stride and padding values change the areas that the kernels will pass through during the convolution operation.

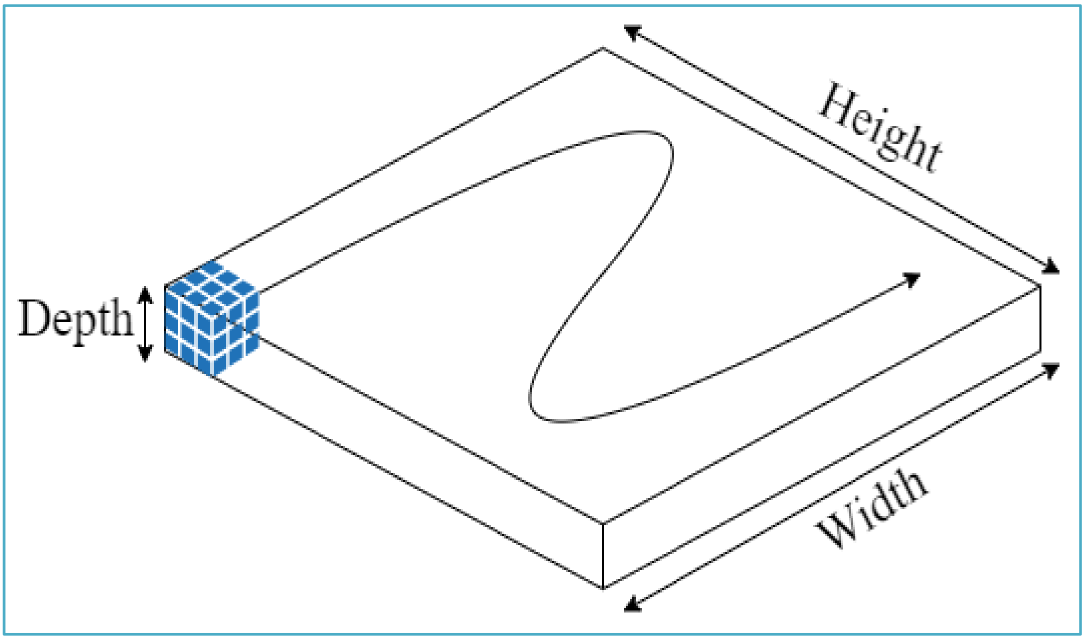


Figure 3.14 : 3D convolution operation with sliding window technique.

The convolution operation shown in Figure 3.14 can be listed as items as follows,

- Flip the mask
- Slide the mask throughout the image
- Multiply the corresponding elements and sum them.
- Repeat it until all values of the image has been convolved.

During the convolution process, the values of the neighbors around the pixels in center are as important as the value of the pixels in center.

The identity, blurring, and edge detection operations given in Section 3.1 can be given as examples of kernel operations.

$$O = \frac{(W - K + 2P)}{S} + 1 \tag{3.4}$$

The dimensions of the new feature map resulting from the convolution operation are shown in Equation 3.4. O represents the output, W is the size of the input, K is the filter size, P is the padding and S is the stride.

Stride Operation:

The process of shifting the kernel matrix over the input matrix is called stride. The critical point is where the filter starts processing and how many steps it takes over the image. The filter moves on the image from left to right and from top to bottom in steps determined by the stride value. If stride is selected as 1, the filter moves 1 pixel at a time. If stride is selected as 2, it will move 2 pixels and so on. The below Figure 3.15 shows the situations where stride is selected as 1 and 2. When 1 is selected, 5x5 output is obtained.

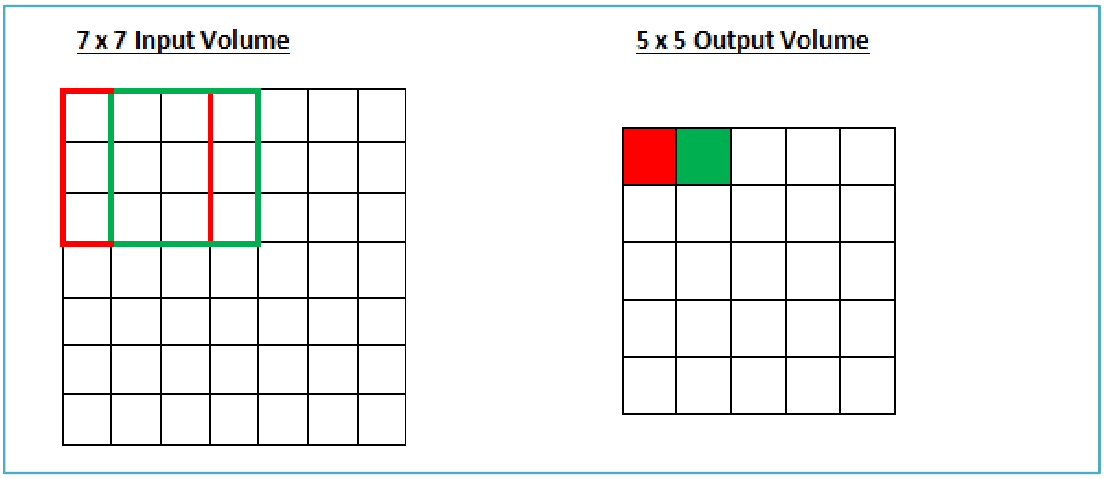


Figure 3.15 : Example of different stride value defining and it’s result(Stride = 1).

As a result of choosing stride as 2, the scrolling process was faster due to the increase in the number of steps and a 3x3 output was created.

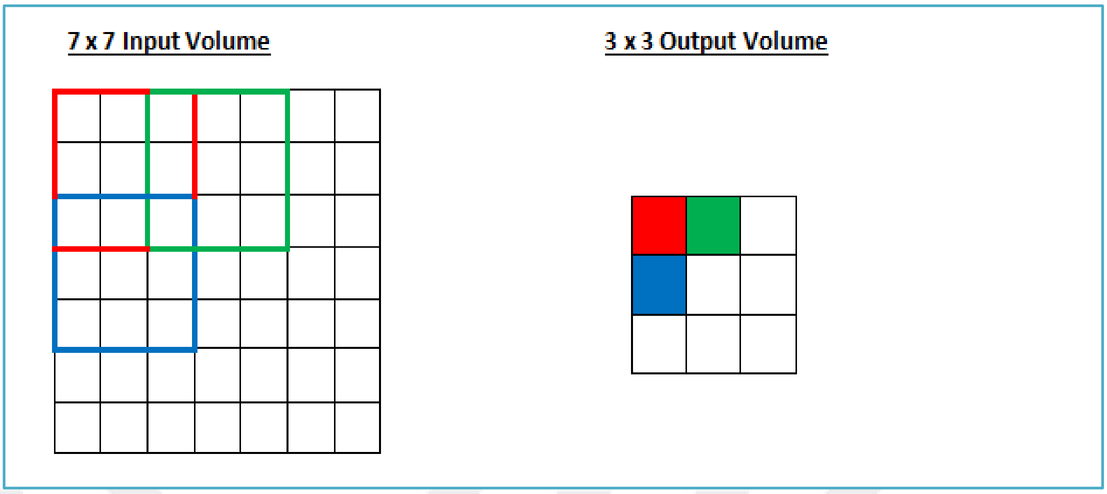


Figure 3.16 : Example of different stride value defining and it's result(Stride = 2).

Stride operation is one of the key components for compression of image and video data in CNN. Since nowadays we are dealing with hardware limitation, it is quite important to have a stride operation which reduces the outputs to smaller sizes, thus enabling a faster training process. The parameters can change depending on how much information we want to retain. We can keep more information in the image with smaller strides. However, with the increase of stride, it becomes difficult to extract meaning from the image. It can be classified as a hyperparameter, as experiments are required to find the optimal value.

Padding Operation:

One of the operations applied to determine the size of the output within the convolution is padding. Spatial size shrinks during the convolution process. If a model has many convolutions, the image may disappear after a certain convolution operation. In such a case, it will be very difficult to create deeper networks. Padding is a process designed as a solution to these problems. It is built by adding 0s around the image at the desired number. In this way, the size difference between input and output is minimized. In fact, in most applications, choosing the padding size as "same" means adding padding so that the dimensions of the input and output are equal to each other. As it is known, another situation that determines the output is the kernel size. If it is large, smaller output will be obtained because the number of scrolling will be less. Currently used programs also decide how much 0 should be added by looking at the kernel size. As stated in Equation 3.5, the number of 0s in "zero" or "same padding" depends on the size of the kernel and K refers to the kernel size.

$$Zero\ Padding = \frac{(K-1)}{2} \tag{3.5}$$

As can be seen in Figure 3.17, adding 1-pixel 0s around the image is shown, meaning that the padding size is selected as 1.

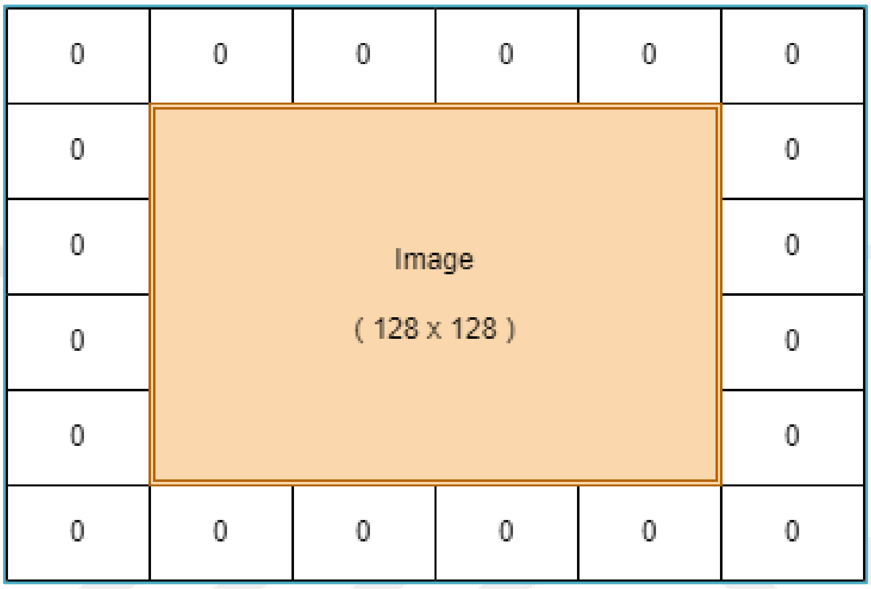


Figure 3.17 : Example of padding operation(zero padding adding to the image).

The important benefits of the padding process can be listed as follows:

- Allows us to build deeper networks
- Ensures the feature map is not lost
- Protects information on borders.
- Protects Spatial size.

There are many types of padding according to the numbers to be added. The most commonly used ones can be listed as follows:

- Zero padding: adding 0s around the image.
- Reflection padding: puts reflections in the borders instead of 0s around the image.
- Replication padding & symmetric padding: copies are placed around the image on the border instead of 0s.

Pooling Operation:

CNN applications were not very common before GPUs because they required very heavy processing power. The reason was that in the feature extraction section, a convolution operation was applied with a large number of filters to extract meaning from the image. After each applied filter, the number of dimensions of the output increases, which means a considerably increased load for the computer after each convolution. For this reason, it was necessary to develop many methods to reduce the processing load. Pooling is one of the methods designed for this purpose. When the amount of convolution is examined in pre-trained models, the pooling process is thought to be very useful.

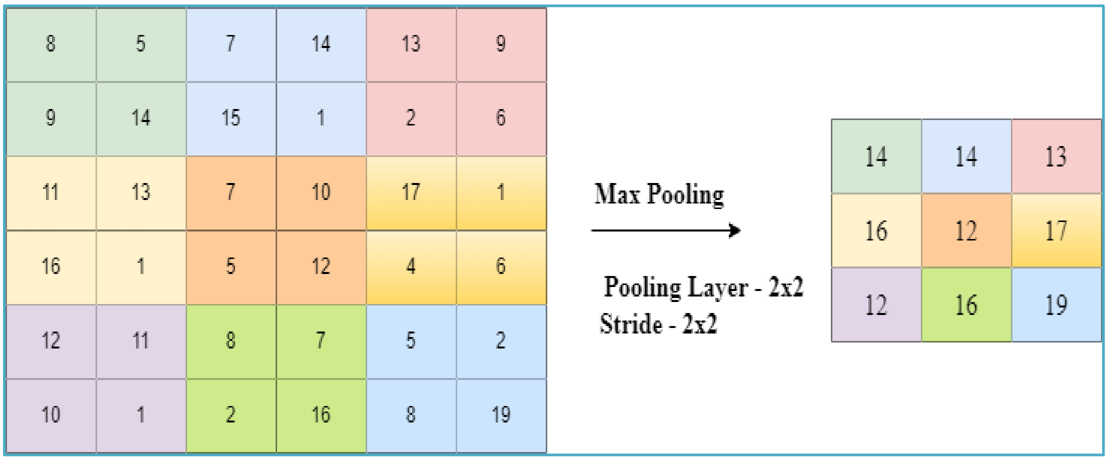
The aim in the pooling process is to get the data that is thought to be the most effective in the determined area in the feature map, so our aim is to give the meaning provided with a much larger number of pixels in a single pixel. There are many methods for pooling in the literature.

The 3 most well-known pooling methods can be explained as follows:

- Average Pooling: All values in the field are averaged.
- Maximum Pooling (Max Pooling): The largest of all values in the field is taken.
- Minimum Pooling: The smallest of all values in the field is taken
- Sum Pooling: The sum of all values in the field is taken.

Pooling matrix size and stride are important during the pooling process. The pooling matrix gives the size of the area to be processed. Stride, on the other hand, tells how many steps the scrolling will take.

Since it is very costly and difficult to operate on CNN without pooling, it is frequently encountered in CNN models. Along with compressing the number of parameters in the model, it adds nonlinearity and prevents the model from being overfitting.



8	5	7	14	13	9
9	14	15	1	2	6
11	13	7	10	17	1
16	1	5	12	4	6
12	11	8	7	5	2
10	1	2	16	8	19

14	14	13
16	12	17
12	16	19

Figure 3.18 : Example of max pooling with 2x2 pooling layer and stride.

In Figure 3.18, the maximum pooling operation to be performed in a 2x2 area is shown. The largest value in the 2x2 fields is taken and since the stride is 2x2, 2 units are shifted horizontally and vertically. In this way, a 6x6 feature map is down-sampled to a 3x3 size. Thus, the dept volume was preserved and the feature map shrunk by 75%, reducing it to 25% of the original size.

Activation Functions:

Activation functions are used in many places in CNN. Activation functions used in the feature extraction section after the convolution or in the transition between layers in the fully connected layer can be given as an example of this situation. The main purpose of their use is to increase the nonlinearity in the network. As it is known, most situations in the world do not work with the linearity rule. In other words, when it is desired to examine a situation, classify or make predictions, linear separations unfortunately do not provide a high accuracy ratio because decision boundaries are not linear. The mathematical equivalent of nonlinearity is that the output cannot be produced as a linear combination of inputs. If the created model is made linear, the added hidden layers cannot contribute to the robustness of the model and it starts to behave like a single layer.

Today, many different activation functions are used in models. In fact, it would be more accurate to classify activation function as a hyperparameter because it is not known which activation function gives better results in the training part. The derivative of the activation function, which is also used during backpropagation, must be available. It is not recommended that there is a problem in the differentiability part, as it will increase the burden on the computer.

In this section, the equations and graphical views of the most frequently used activation functions are shared.

Sigmoid Activation Function: It is one of the most frequently used activation functions with a range of 0 to 1. It is a type of activation function that is frequently encountered in binary systems. Since the result converges to 0 or 1 after a certain value, the model has difficulty in learning.

The graph of the derivative and itself is shown in Figure 3.19.

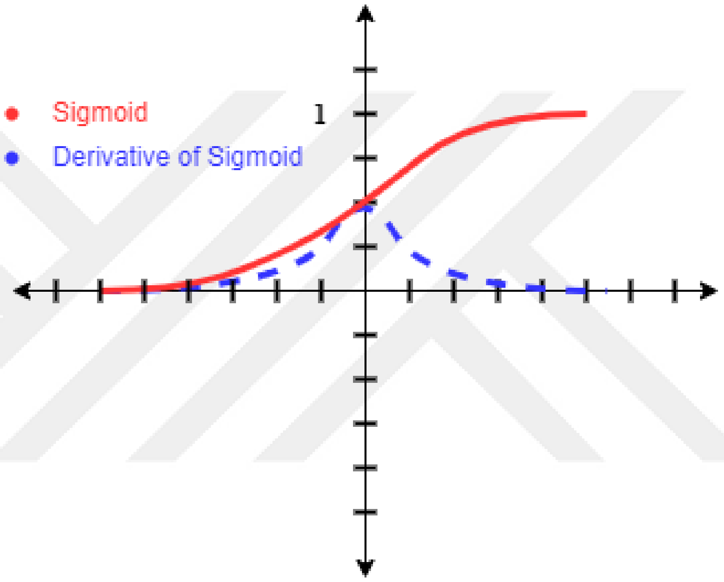


Figure 3.19 : Graph of Sigmoid and the derivative of the Sigmoid function.

The equation of the sigmoid function is given,

$$\sigma(x) = \frac{1}{1 + \exp(-x)}, \qquad 0 < \sigma(x) < 1 \tag{3.6}$$

When its derivative is calculated, the point with the highest value is the point where it is closest to 0. The derivative equation is as follows.

$$\sigma'(x) = \sigma(x)(1 - \sigma(x)) \tag{3.7}$$

Tanh Activation Function: It is similar to the sigmoid function, but the boundaries are not limited to the positive area, it is possible to have negative values. It is stated that it works better than sigmoid, but since the change after a certain interval is very small as in sigmoid, vanishing gradient problem is also present in tanh.

The graph of its derivative and itself is given in figure 3.20.

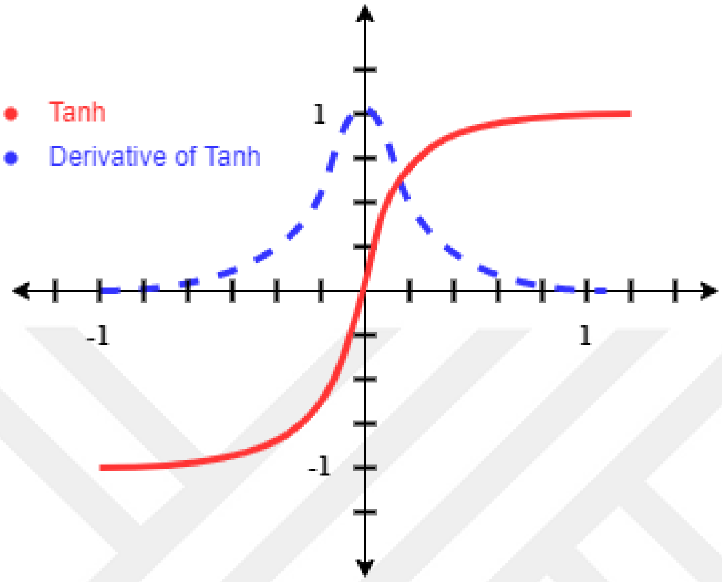


Figure 3.20 : Graph of Tanh and the derivative of the Tanh function.

The equation of the Tanh function is given,

$$tanh(x) = \frac{1 - exp(-2x)}{1 + exp(-2x)}, \qquad -1 < tanh(x) < 1 \tag{3.8}$$

The equation of the derivative of the Tanh function is given,

$$tanh'(x) = 1 - \tanh^2(x) \tag{3.9}$$

Rectified Linear Units (ReLU) Activation Function: The processing load on the computer is quite low compared to other activation functions. For this reason, it is one of frequently preferred activation functions. In addition, the models generally give better results in comparisons, making it the most frequently used activation function. However, the absence of a gradient value in negative values causes the problem of vanishing gradient in backpropagation. For this reason, the Leaky ReLU function, which will be mentioned in the 4th item, has been evolved to be a solution to this mentioned problem of ReLU.

The graph of its derivative and itself is shown in Figure 3.21.

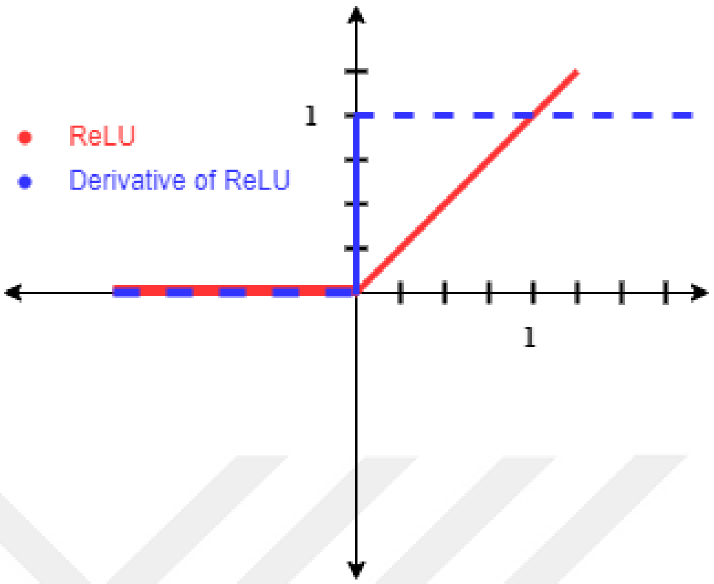


Figure 3.21 : Graph of ReLU and the derivative of the ReLU function.

The equation of the ReLU Function is given,

$$ReLU(x) = max(0, x) \ , \ \ 0 \leq ReLU(x) \qquad (3.10)$$

The equation of the derivative of the ReLU function is given,

$$ReLU'(x) = \begin{cases} 0 & if \ x \leq \ 0 \\ 1 & if \ x > \ 0 \end{cases} \qquad (3.11)$$

Leaky Rectified Linear Units (Leaky ReLU) Activation Function: It shows us that the result will not change as x changes on the negative side of ReLu. For this reason, it was mentioned that what kind of problem would cause in the previous section. This issue has been resolved with a user-specified slope with a ratio of “a” on the negative side.

The graph of its derivative and itself is shown in Figure 3.22.

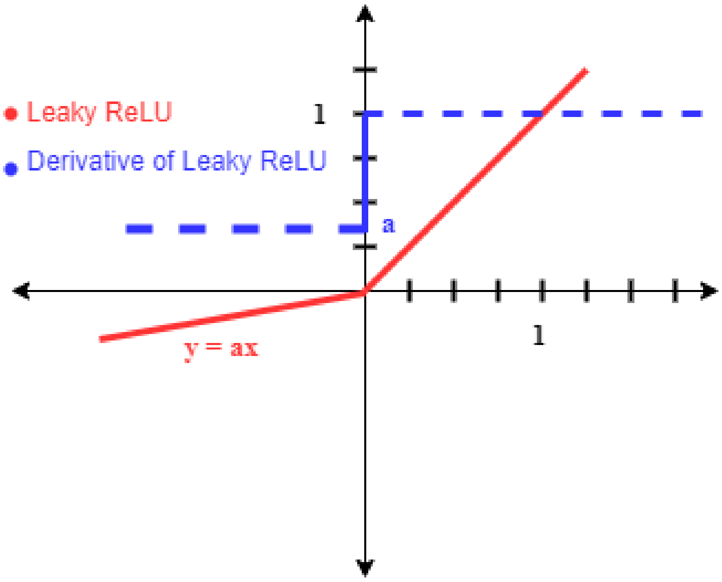


Figure 3.22 : Graph of Leaky ReLU and the derivative of the Leaky ReLU function.

The equation for the Leaky ReLU Function is given,

$$Leaky\ ReLU(x) = \begin{cases} x & if\ x \geq\ 0 \\ ax & if\ x <\ 0 \end{cases} \tag{3.12}$$

The equation of the derivative of the Leaky ReLU function is given,

$$Leaky\ ReLU'(x) = \begin{cases} 1 & if\ x >\ 0 \\ a & if\ x <\ 0 \end{cases} \tag{3.13}$$

Flattening Operation:

Flattening Layer is located between feature extraction and cnn layer and connects these two layers. It is basically called the flattening process before entering the cnn layer, because the input of the neural network must be a 1-dimensional linear vector. Since the inputs in different forms (rectangular, cubic) and different dimensions cannot be taken directly to the FCNN layer, they are brought into 1-dimensional form in the flattening layer. The feature maps created as a result of the operations made in the feature extraction layer have many dimensions and a flattening layer is needed because many dimensional forms give an error when they are sent to the FCNN layer.

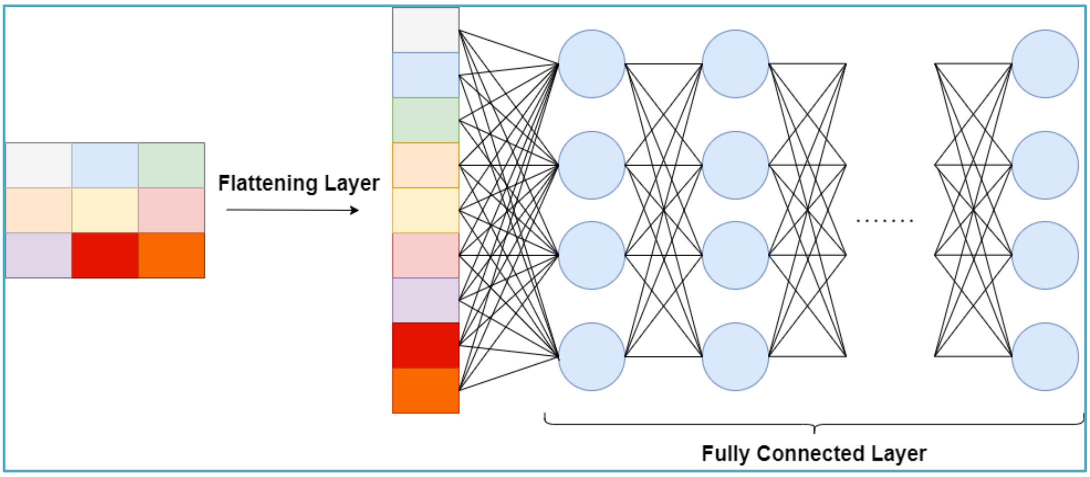


Figure 3.23 : Flattening operation and its FCNN connections.

Fully Connected Neural Networks:

It is the next layer after feature extraction and flatten layer in CNN. As the term fully suggests in its name, neurons in each layer are connected to all neurons in the next layer. The number of hidden layers varies from model to model. Regardless of the number of neurons in the hidden layers, the number of neurons usually decreases towards the end layer and reaches the output.

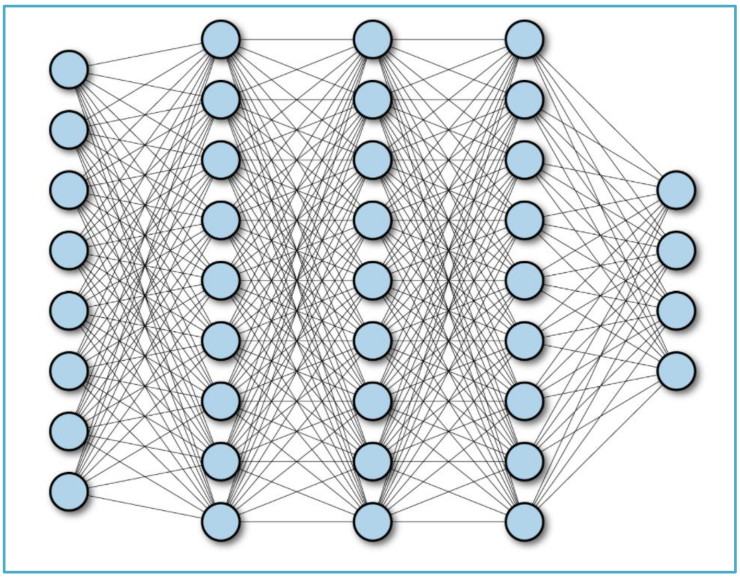


Figure 3.24 : Fully Connected Neural Network having 3 hidden layers.

Activation functions are also available here. Each node transmits the values of (previous nodes * weight between two nodes + bias) to the the activation unit and the result of becomes the weight of its own node. The resulting weight is expressed with y and the equation is formulated as in 3.14.

$$y = \sum_{i=1}^{m} w_i x_i + b \qquad (3.14)$$

$$output = \sigma\ (\mathrm{y}) \ \ , \ \sigma \text{ is activation function}$$

The results of each node are transmitted to the output layer by repeating the above equation recursively.

Classification Layer:

It constitutes the last layer of the model and is the part where the prediction is made.

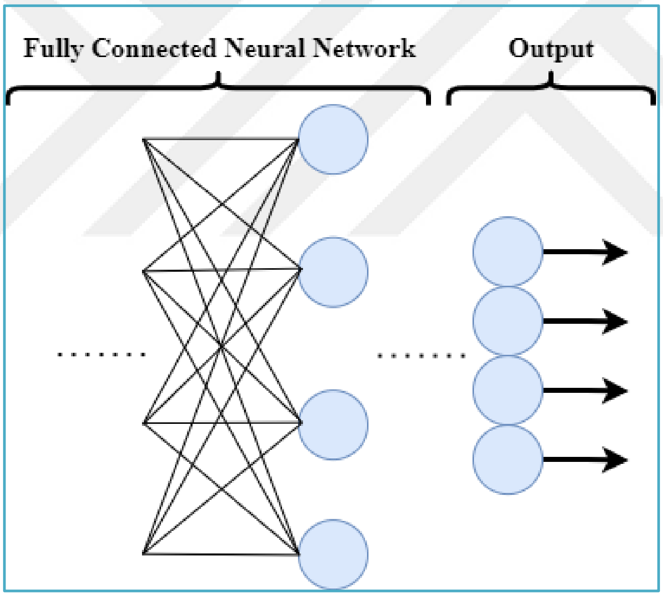


Figure 3.25 : Last layers of CNN (FCNN and output layers).

In models that perform the classification process, usually the data received after the output layer of the model is converted into probabilistic values with the Softmax function and then the amount of error is tried to be measured with Cross-Entropy loss.

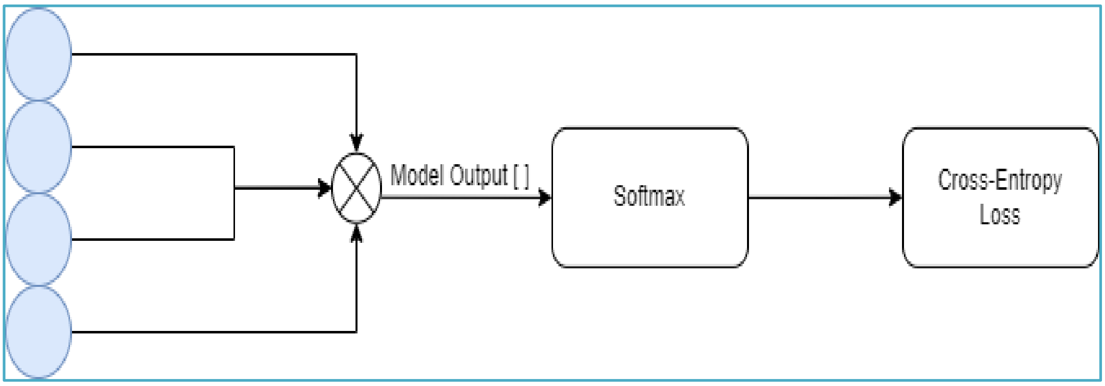


Figure 3.26 : Softmax transformation of output vector and sending them to the Cross-Entropy section for loss measurement.

The formulation of Softmax is shown in Equation 3.15.

$$Softmax(S)_i = \frac{e^{S_{i,}}}{\sum_j^C e^{S_j}} \tag{3.15}$$

The loss between the actual values the results coming from Softmax found by L1, L2 or Cross-Entropy loss and by starting the back-propagation process, the loss is tried to be minimized. The resulting loss is given in Equation 3.16

$$Loss_{cross\_entropy} = -\sum_i y_i \log(Softmax(S)_i) \tag{3.16}$$

Since Cross-Entropy loss is used very frequently in classification problems, it was felt the need to explain in more detail in this section.

3.3 Learning in Deep Neural Networks

Loss Function:

Backpropagation is based on how wrong the model's decision is. The amount of change in weights is determined by the amount of error calculated by the various loss functions.

- L1 Loss(Mean Absolute Error) Function:

The L1 Loss function is created as the sum of the absolute values of the errors.

$$Loss_{l1} = \sum_{i=1}^{n} \left| y_{true} - y_{predicted} \right| \quad (3.17)$$

- L2 Loss(Mean Square Error) Function:

The L2 loss function is created as the sum of the squared values of the errors.

$$Loss_{l2} = \sum_{i=1}^{n} \left(y_{true} - y_{predicted} \right)^{2} \quad (3.18)$$

- Root Mean Square Error:

The impact of the change made in RMSE is less compared to L2 because taking the square root causes the error to be standardized.

$$Loss_{root\_mean} = \sqrt[2]{\sum_{i=1}^{n} \left(y_{true} - y_{predicted} \right)^{2}} \quad (3.19)$$

- Cross-Entropy Loss:

It is a type of loss frequently used in classification problems. It is often used in conjunction with Softmax. The conversion of the values from the model to the probability value is done with Softmax, which was explained in Section 3.2. Then the loss function is created with the difference between the actual and the predicted value.

$$Loss_{cross\_entropy} = -\sum_{i} y_i \log(\hat{y}_i) \quad (3.20)$$

Backpropagation:

Backpropagation is a method of minimizing the error applied in supervised learning models of Artificial Neural Networks. The gradient of the error amount obtained as a result of the selected loss function shows us the direction that will increase the error the most. The opposite of this direction is the direction in which the error will decrease the most. In this way, the training of the models is provided with changes that will minimize the error.

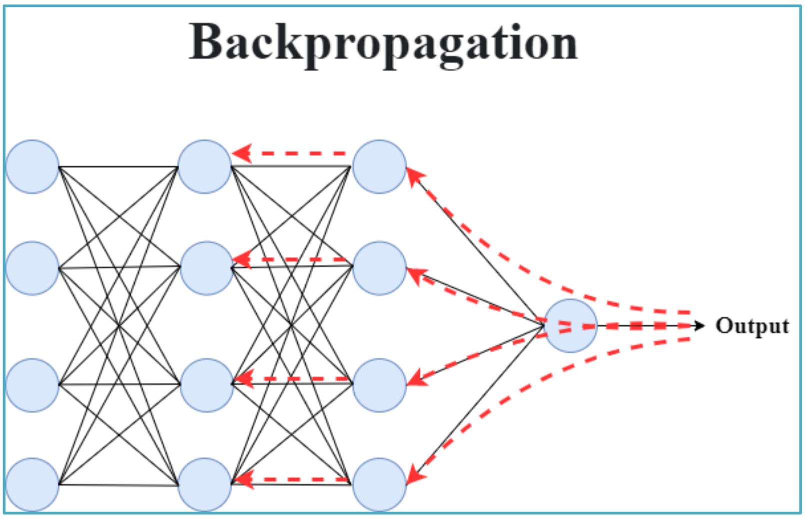


Figure 3.27 : Backpropagation of the model.

As shown in Figure 3.27, backpropagation proceeds from back to front, and optimization methods with varying weights of parameters are updated by using derivatives of the equations connecting the model.

$$\Delta w_{i,j} = -\alpha \frac{\partial E}{\partial w_{i,j}} \tag{3.21}$$

The process of updating the weights, whose generalized equation is given in Equation 3.21, is the main goal during backpropagation.

Optimization:

Optimization is the set of mechanisms that decide how large the weights of the model will change during backpropagation. With this method, which directly affects the learning coefficient (ℓ), the weights of the neurons included in the model change and it is expected to make more accurate predictions. It updates the learning rate during backpropagation with the error rate it received during training and hopes to make fewer errors in the next batch. At this point, the gradient of the network is very important. The derivative used in the gradient calculation shows the direction in which the error rate increases the most. By going in the opposite direction of this increase, the current error amount is reduced. The right optimization method will make learning faster, which makes it possible to save time. In this respect, the optimization method to be used is very important. The most commonly used optimization methods are SGD, Momentum and Adam.

Stochastic Gradient Descent (SGD): The updating of the weights as a result of multiplying the gradient value obtained by backpropagation with the determined learning rate is called SGD and it is often confused with Batch Gradient Descent. The only difference is that while the cost of the entire training dataset is calculated in BGD, the cost of the example selected in each step in SGD is taken, not the entire dataset.

$$\theta_j = \theta_j - \eta . \nabla_\theta J(\theta; x, y) \tag{3.22}$$

Momentum: Momentum emerges as a more advanced form than SGD. In addition to the equation, γ called momentum and vt parameters are added. The momentum variable is given to the system as a constant and vt represents the change in the previous weight update.

$$\theta_j = \theta_j - \eta . \nabla_\theta J(\theta; x, y) + \gamma v_t \tag{3.23}$$

Adam: The Adam method, which has become very popular in recent years, has features such as computationally efficient and less memory requirement. As it is known, problems such as staying at the local minimum or not being able to catch the minimum due to the large learning rate may cause a bad training process but adaptively changing the constants in Adam makes the converge more robust.

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{3.24}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{3.25}$$

m_t and v_t can be explained as the previously stored momentum and gradient values. These values will also be used to determine what the next weight will be. The values of β_1 and β_2 are defined as the decay rate and are close to 1.

$$\widehat{m_t} = \frac{m_t}{1 - \beta_1^t} \tag{3.26}$$

$$\widehat{v_t} = \frac{v_t}{1 - \beta_2^t} \tag{3.27}$$

Afterwards, the bias correction process is specified in equations x and y.

$$\theta_j = \theta_j - \frac{\eta}{\sqrt{\widehat{v_t} + \epsilon}} . \widehat{m_t} \tag{3.28}$$

The equation on how to change the weights to be updated as a result of the values (m_t, v_t, β_1, β_2) calculated above is given.

In this thesis study, all optimization methods have been tried and Adam algorithm has been found to give the best results, so Adam has been chosen as the optimizer.

Dropout Regularization:

Dropout is one of the regularization methods frequently used in deep learning models. It provides a healthy learning process that will prevent the model from being overfitting. In order to achieve this situation, randomly selected neurons are forgotten and learning does not occur in this neuron. In this way, the effect of ignored neurons on the model is temporarily stopped, allowing other neurons to learn. Because if the network is very large or the training process takes too long, there is a risk of overfitting the model so the model will show much less success during the testing than it did during training. When dropout is applied, the connection between neurons is ignored not only in forward propagation but also backward.

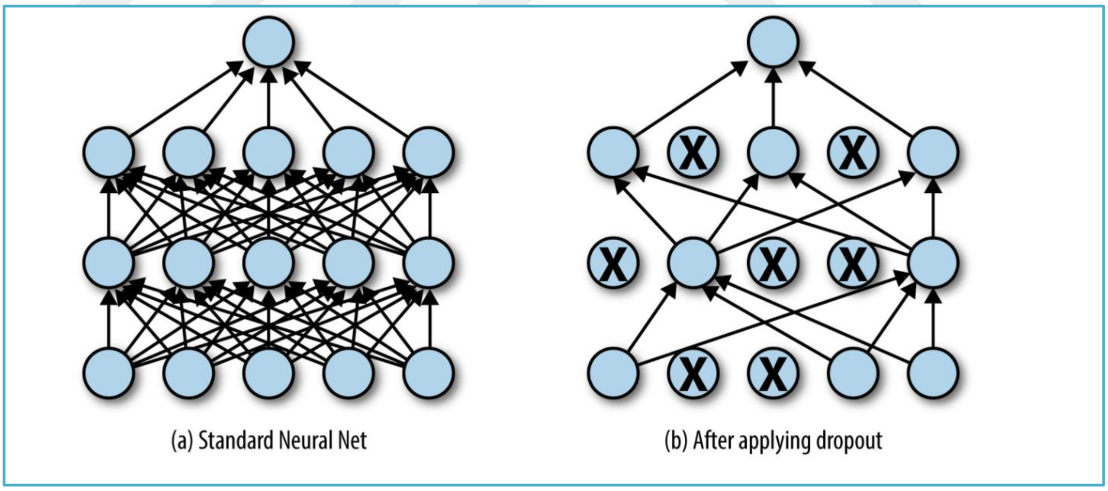


Figure 3.28 : Comparison of standard model and model having dropout.

As indicated in Figure 3.28, the normal fully connected neural network is seen on the left side. On the right, the connections that occur after adding the dropout are shown. How many neurons will be ignored is one of the hyperparameters. Therefore, the user has to determine it. It is usually stated as p in the literature, it is called the keeping ratio and it should take a value between 0 and 1. A value of 1 means that the network is learning in a standard way, so no neurons are ignored. If it is 0, all neurons in the network are ignored. Generally applied are values of 0.5 and above. In addition, dropout is only used during the training in the model, and all neurons are connected to each other during the testing.

3.4 Divergence-Based Convolutional Neural Network

In this section, the concept of divergence-based, which is the source of motivation for the study, and why it is preferred will be mentioned. At the same time, the structure of the proposed model will be shared and the applicability of the proposd method in the CNN model will be shown. Then, Walsh Matrix will be discussed and the properties of this matrix, which is preferred for class representation, will be examined.

3.4.1 Divergence based networks

The divergence value calculated over the feature set measures the intra-class representativeness of the features and the effect value of being discriminatory out of the class. The goal in training is to have a high divergence value. An within-class scatter matrix is needed to measure the divergence value. This means the sum of the covariance matrices of all classes. Equation 3.29 shows the equation for creating the within-class scatter matrix[53].

$$\mathrm{S} = \mathrm{S}_1 + \mathrm{S}_2 + \cdots + \mathrm{S}_k\,,\; S_n \;=\; covariance\, matrix\, of\, nth\, class \tag{3.29}$$

In Equation 3.30, the equation for the creation of divergence value is given. B represents between-class scatter matrix.

$$divergence\, value \;=\; tr(S^{-1}\, \cdot\, B) \tag{3.30}$$

According to the equations shared above, the large ratio of the between-class scatter causes a higher divergence value, which is important in terms of distinguishing the classes. In this way, the classes can be placed at the furthest points in the feature space. The vectors that will provide this activity are Walsh Matrixes, which will be explained in the next section[53].

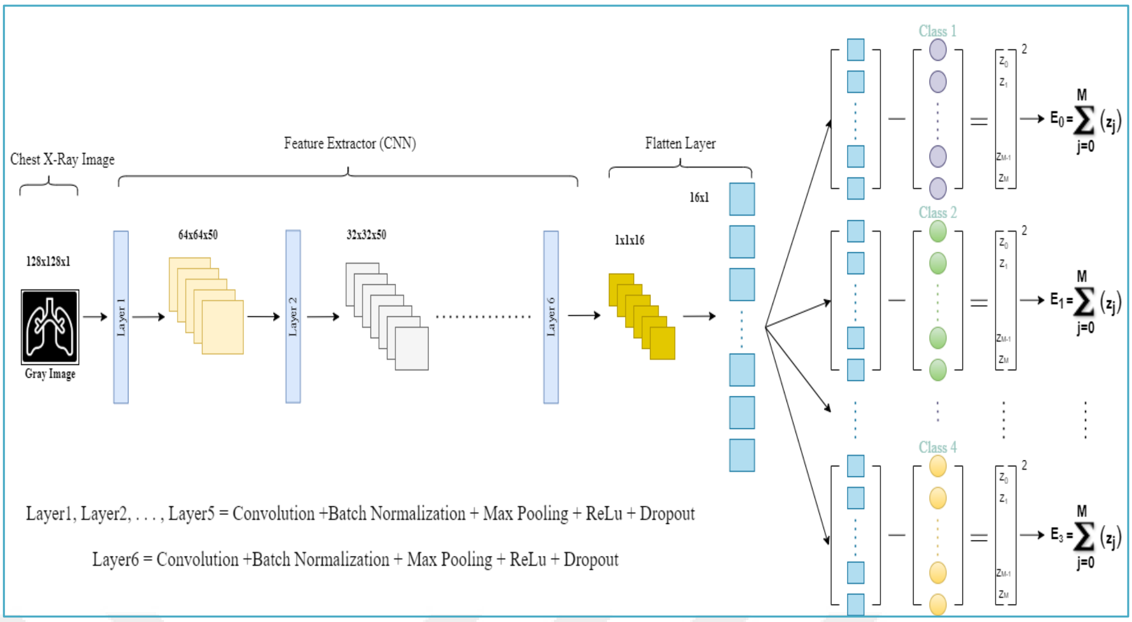

Chest X-Ray Image
Feature Extractor (CNN)
Flatten Layer
128x128x1
Gray Image
Layer 1
64x64x50
Layer 2
32x32x50
Layer 6
1x1x16
16x1
Class 1
Class 2
Class 4
z_0
z_1
z_{M-1}
z_M
2
$E_0 = \sum_{j=0}^{M} (z_j)$
$E_1 = \sum_{j=0}^{M} (z_j)$
$E_3 = \sum_{j=0}^{M} (z_j)$
Layer1, Layer2, . . . , Layer5 = Convolution +Batch Normalization + Max Pooling + ReLu + Dropout
Layer6 = Convolution +Batch Normalization + Max Pooling + ReLu + Dropout

Figure 3.29 : Proposed model structure.

The structure of the Proposed model is shared in Figure 3.29. As it can be seen, the layers in the feature extraction part are not different from other CNNs. The class with the least difference is determined by subtracting the 16x1 vectors coming from the Flattening Layer and the Walsh vectors representing the classes and it is claimed that the model predicts that class.

3.4.2 Learning by Walsh vector

The Walsh function is a matrix formed by the combination of orthogonal vectors. For this reason, it is called an orthogonal function. It can be created in continuous vs discrete forms. When Walsh matrix is created in discrete form, it contains only -1 and 1 values. The dot product is 0 because the vectors it contains are perpendicular to each other[54], [55].

The Walsh Matrix was proposed by Joseph L. Walsh. Each row of the walsh matrix is also referred to as a walsh function[56].

It is seen that the Walsh Matrix is derived from the Hadamard Matrix. It is shown in Equation 3.31 of the Hadamard Matrix.

$$\mathrm{H}_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad (3.31)$$

The number 2 in the H_2 notation indicates that it is a 2x2 matrix and can be expanded. As shown in Equation 3.32, the transposes of these matrices are also equal to each other.

$$\mathrm{H}_2 = \mathrm{H}_2^{\mathrm{T}} \tag{3.32}$$

Hadamard matrix with expandable form were increased to 16x16 by recursive method in this study. The required operation for expansion is shown below with Equation 3.33.

$$\mathrm{H}_2^{\mathrm{k}} = \begin{bmatrix} H_2^{k-1} & H_2^{k-1} \\ H_2^{k-1} & -H_2^{k-1} \end{bmatrix} \tag{3.33}$$

Then, using the above equation, H_4, H_8 and H_{16} were formed recursively. In addition, since Relu will be used in the deep learning model created, the possible data will have positive value. Therefore, -1 values are replaced by 0 in the matrix created and they are named W_4, W_8, W_{16} [53].

$$\mathrm{W}_2 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix},\ \mathrm{W}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix},\ \mathrm{W}_8 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix} \tag{3.34}$$

Then, by using the W_8 matrix and repeating the operations with the rules in Equation 3.31, the 16x16 matrix that we will use in our model is obtained.. As can be seen, it is possible to create Walsh Matrixes that can be brought into the desired format with the rule in Equation 3.31. The matrix formed in Equation 3.35 is shown below.

$$W_{16} = \begin{bmatrix} 1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1 \\ 1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0 \\ 1&1&0&0&1&1&0&0&1&1&0&0&1&1&0&0 \\ 1&0&0&1&1&0&0&1&1&0&0&1&1&0&0&1 \\ 1&1&1&1&0&0&0&0&1&1&1&1&0&0&0&0 \\ 1&0&1&0&0&1&0&1&1&0&1&0&0&1&0&1 \\ 1&1&0&0&0&0&1&1&1&1&0&0&0&0&1&1 \\ 1&0&0&1&0&1&1&0&1&0&0&1&0&1&1&0 \\ 1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0 \\ 1&0&1&0&1&0&1&0&0&1&0&1&0&1&0&1 \\ 1&1&0&0&1&1&0&0&0&0&1&1&0&0&1&1 \\ 1&0&0&1&1&0&0&1&0&1&1&0&0&1&1&0 \\ 1&1&1&1&0&0&0&0&0&0&0&0&1&1&1&1 \\ 1&0&1&0&0&1&0&1&0&1&0&1&1&0&1&0 \\ 1&1&0&0&0&0&1&1&0&0&1&1&1&1&0&0 \\ 1&0&0&1&0&1&1&0&0&1&1&0&1&0&0&1 \end{bmatrix} \qquad (3.35)$$

The transformation from Hadamard Matrix to Walsh matrix takes place by interchanging the rows. This is accomplished by converting from binary code to gray code. In other words, assuming that there is an order of Row0, Row1, Row2 and Row3 in a 4x4 array, it is arranged as Row0, Row2, Row3, Row1 after the Gray code transformation, which means that Row1 is positioned in the last row. Since the Walsh matrix order is not important for us in this study, this transformation may not be done, but it may be necessary in the communication systems[57].

The reason for creating a 16x16 matrix is that we have a 16x1 array when the image come to the flatten layer from the convolutional neural network, as can be seen in Figure 3.20. If necessary arrangements are made (kernel size change, convolution number change), the size of the usable walsh matrix can be changed.

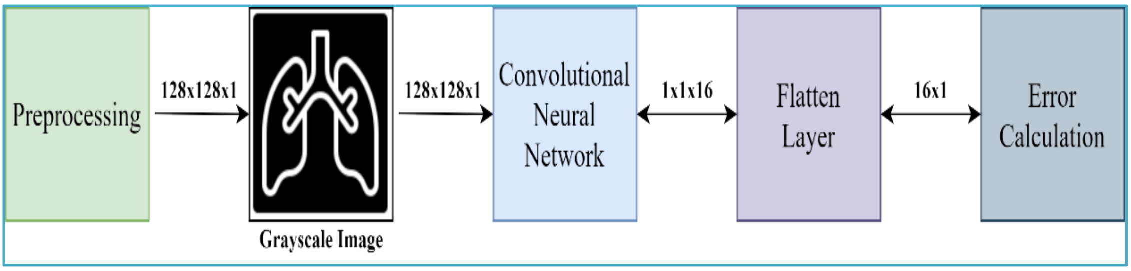


Figure 3.30 : General layers of proposed model.

The purpose of using Walsh Functions is to create the farthest points on the spatial plane. In this way, by positioning the classes as far away as possible, the model will be able to classify more accurately. These vectors with 0 dot products with each other were used in this study. As can be seen in Figure 3.31, the 4 vectors taken from the 16x16 matrix correspond to the 4 classes attempted to be classified. Since the vectors in the Walsh matrix are at the farthest point from each other, 4 of them can be chosen randomly.

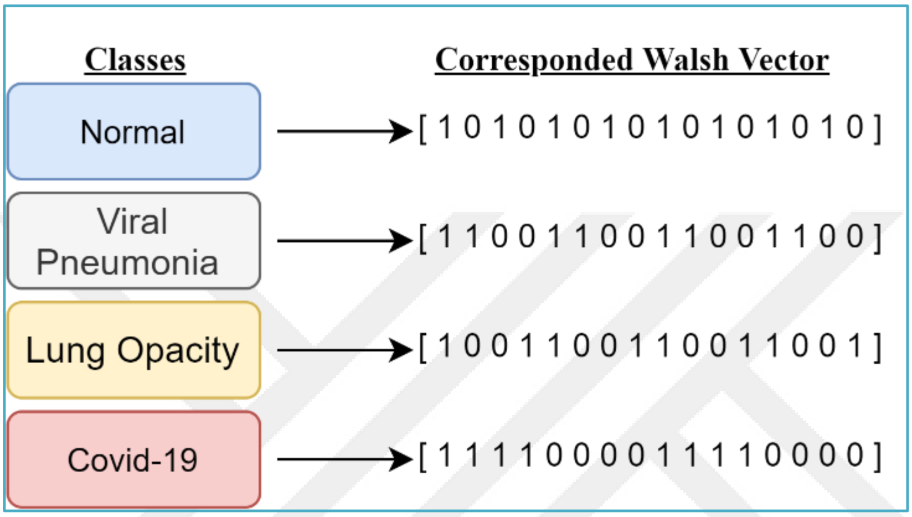


Figure 3.31 : Classes and corresponding Walsh vectors.

3.5 Classification of Chest X-Ray Images

As stated in the previous sections, various imaging methods are used to determine the lung condition. Mainly used methods are chest CTs and CXRs. Although CXR is less sensitive in various researches compared to CT Scan, it is used more frequently because it contains less radiation and is more accessible. Therefore, the amount of data generated is relatively higher.

In this study, X-ray images were preferred in terms of data accessibility, reliability and amount of data. Choosing a public dataset also has an importance in terms of being accessible to all other researchers.

Classification has been done by physician for years. Specialized radiologists look for evidence to make a diagnosis by looking at the images taken.

In general, the color of the lungs and surrounding organs and tissues tells us whether there is a problem.

Colors present in chest radiography are interpreted as follows:

- White Colour: Bone /Calcium or calcified structures (e.g., ribs)
- Gray Color: Structures consisting of soft tissue (e.g., heart)
- Dark grey: Fat Tissue
- Black Color: Structures containing air (e.g., lung and intestine)
- Deep Black: Air & Gas

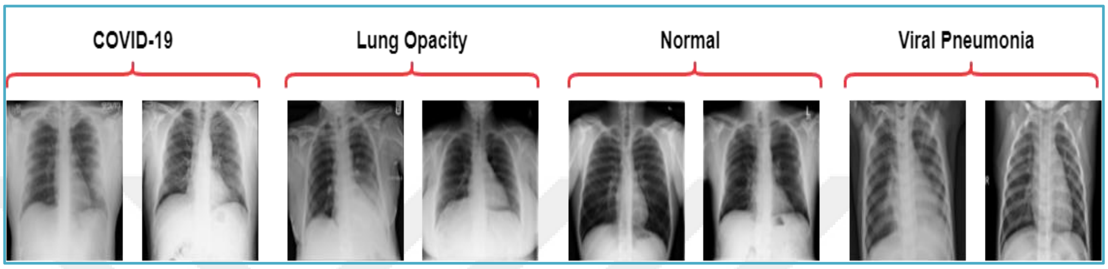


Figure 3.32 : Images of each class in the dataset.

Colors other than the ones specified show us that there may be problems in that area. When we look at the images with Covid-19, especially the lung region should be dark black, but the color close to white shows that the oxygen saturation decreases. For this reason, breathing devices are used to increase the oxygen saturation of the patients.

4. COMPUTER SIMULATION

In this section, the dataset, performance measurements and the differences of the proposed model with other pre-trained models will be examined. Comparisons will be interpreted on the weights of the models, hyperparameter needs, training times and the performance obtained. Programs required for testing are written in Python. The results and times obtained may vary depending on the programming languages, the starting weights of the models, and the hardware used. In addition, the randomness that exists in the whole system may also affect the results.

4.1 Dataset of Chest X-Ray Images

This is the dataset that many universities (Qatar University, University of Dhaka) and country (Pakistan, Malaysia) teams created together with medical doctors and shared on Kaggle [58] used in the study. This thesis study has been continued with the images in Version 4, which has been updated several times. The received data set contains 4 different classes. Classes are named Covid-19, Normal, Lung Opacity and Viral Pneumonia. As can be seen in Figure 4.1, the dataset has a total of 21,165 images and its total size is around 776 Mb. The data numbers for the classes are 3,616 Covid-19, 6,012 Lung Opacity, 10,192 Normal, 1,345 Viral Pneumonia.

In addition to not all images are clear, there are images obtained with different imaging methods (CT scan) and without a clear view of the chest area. After cleaning such images, which can also be called noise, there are serious differences in the number of images included in the classes. In artificial intelligence-based applications, there should not be such a difference in the number of variables between the classes. However, when this dataset is examined, it includes Normal 10,192 and Viral Pneumonia 1345 images, which means a difference of about 7.5 times. Another requirement for Deep Learning models not to overfit is that the classes contain equal or close to equal numbers of data. For this reason, the differences between the augmentation techniques and the number of images of the classes have been reduced, making them as indicated in Figure 4.1.

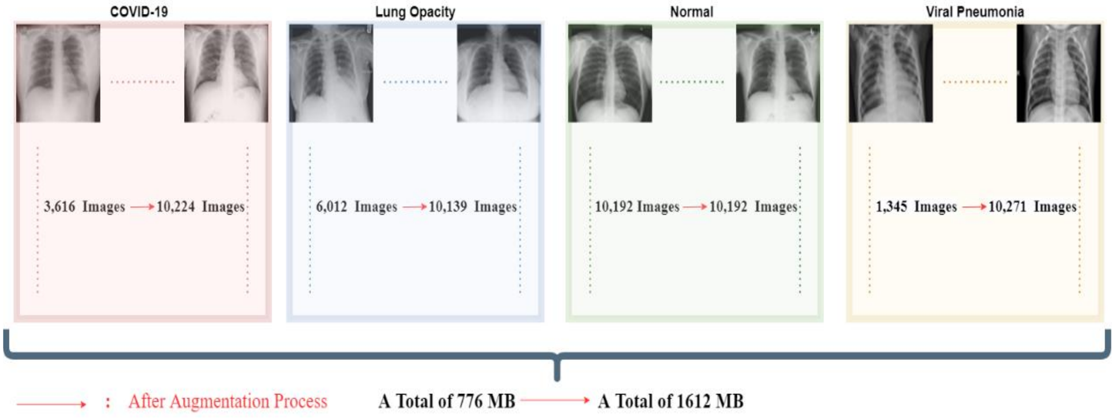

COVID-19
Lung Opacity
Normal
Viral Pneumonia
3,616 Images → 10,224 Images
6,012 Images → 10,139 Images
10,192 Images → 10,192 Images
1,345 Images → 10,271 Images
→ : After Augmentation Process
A Total of 776 MB → A Total of 1612 MB

Figure 4.1 : Number of image in the Chest X-Ray dataset before and after augmentation.

After the image numbers are balanced, the data must be divided in various proportions. Because different images must be used during model training and testing. In addition, a validation set should be created to examine the status of success rates during model training. As a result, there must be 3 different clusters and these clusters must contain different proportions of images. As can be seen in Figure 4.2, the images are shared as 70%, 20%, 10% in the train, test and validation clusters.

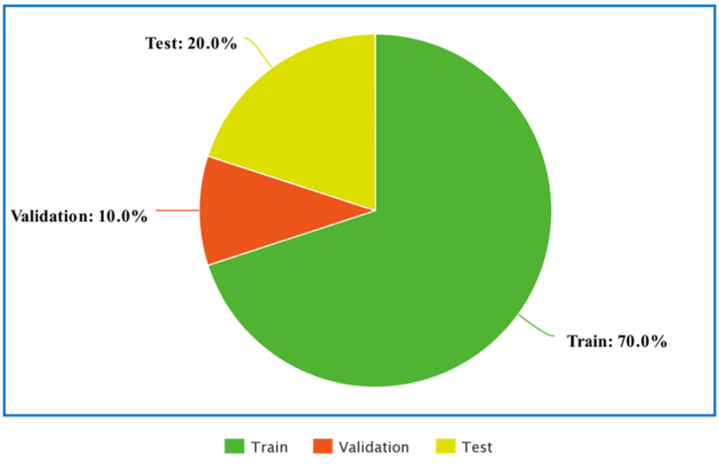


Figure 4.2 : Split rates of the dataset into train, test and validation.

The environment in which programs run is of critical importance. The resources provided by UHeM were used during the creation of the model. The skeleton of the proposed model was created with these resources, and then the proposed and pre-trained models were compared with the test computer in terms of time and success. The reason for using two different systems is to create models that enable training with low-power computers, which is the purpose of this thesis. For this reason, since the resource provided by UHeM is difficult to access, the models were trained with a personal computer for comparison and the results were shared.

Table 4.1 : The properties of test computer.

Properties of Test Computer	Value
Processor	i7-10750H
GPU	RTX 2060
Multi GPUs	No
Cuda Version	10.2
Python Version	3.8.11
Memory	32 GB
Frequency of Memory	2933 Mhz
Running Environment	Conda/Spyder
Cpu Cores	6

The libraries used in the project are OpenCv, Pandas, Numpy, Torch and Tensorflow. Apart from these libraries, there are other libraries (sys, math, time, json etc.) coming from built-in modules with the environment created.

Dimensions of all images may differ from each other. Therefore, all images have been converted to gray level. OpenCV was used for these operations and it was formatted as 128x128x1 regardless of its dimensions. Afterwards, it was sent to the pre-processing specified in Section 3.1 in order to reduce overfitting and increase the model success. In this part, TensorFlow, Pandas and NumPy libraries were used. After the preparation was completed, the images were transferred to the model. The functions required for modeling are done by TensorFlow and PyTorch libraries. It has been tried to make maximum use of the libraries that PyTorch has provided.

This model, which was developed as a result of these studies, will be analyzed by comparing it with other pre-trained models in terms of the number of variables and training times. With the results, the good and bad sides of this model structure will be specified. The results obtained by comparing the performance analysis of the networks in terms of the number of weights and training times will be shared in the next parts of Section 4 and in Section 5.

4.2 Classification Performance Measures for Chest X-Ray Images

Performance measurements are a very important issue in all areas of artificial intelligence. Compared to other fields, classification problems can be measured relatively easily. Because there are sharper boundaries between right and wrong. On the other hand, classification achievements should include not only accuracy but also extra values such as precision, recall, F1-Score. Otherwise, it cannot be known how well the model is and the measured values should belong to the test dataset. In other words, after the model is trained, the success of the model is measured with the test set.

The first thing to do with model’s performance measurement is to create a confusion matrix. Thanks to this matrix, the comparison of whether the model prediction belongs to TP, FP, FN, TN can be observed. As can be seen in Figure 4.3, in many studies, confusion matrix contains only 2 different classes and as a result of these ratios, performance graphics are created. The most frequently used terms in the literature for achievement comparisons are as follows.

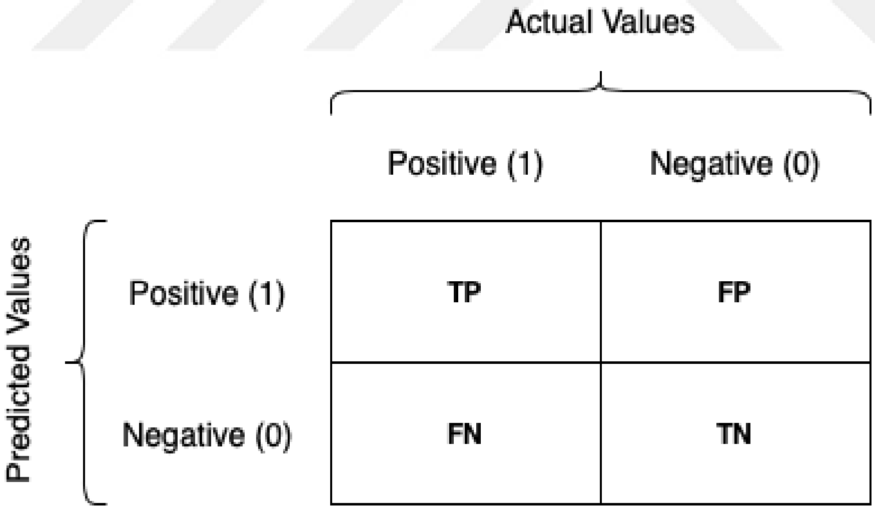


Figure 4.3 : Confusion matrix with 2 classes.

Accuracy: It shows proportionally how many samples is correctly classified by the model. Correct predictions of the confusion matrixed model are shown as TP and

TN. The main purpose of the model is to keep the TP and TN values at the highest level.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.1}$$

Precision: It is how many of the predicted positive samples are predicted correctly, and the False Positive value is very important.

$$Presicion = \frac{TP}{TP + FP} \tag{4.2}$$

Recall: It shows how many of the actually positives did the model correctly identify, and the False Negative value is very important.

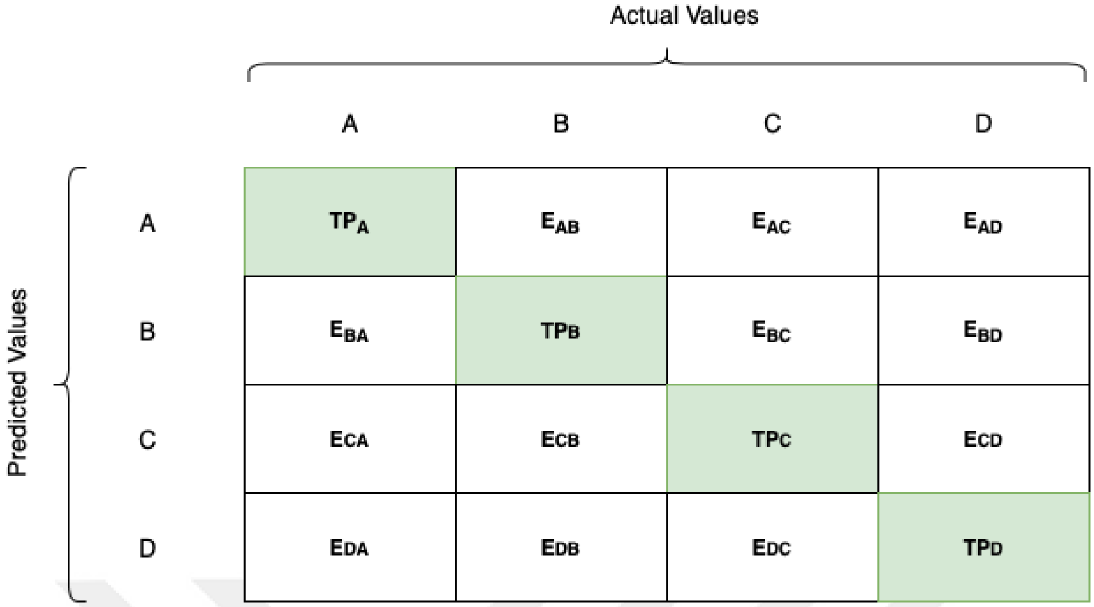
$$Recall = \frac{TP}{TP + FN} \tag{4.3}$$

F1-Score: It is used for values where False Negative and Positive values are significant. It is preferred metric for imbalanced data.

$$F1\,Score = 2 * \frac{presicion * recall}{presicion + recall} \tag{4.4}$$

AUC-ROC Curve: This two-axis graphic, which is referred to as AUC-ROC in the literature, is frequently used to measure the success of the model in classification. In the ROC graph, the x-axis shows False Positive Rate and the y-axis shows True Positive Rate. It provides an overview of the model's performance in the area under the curve (AUC). The larger the area covered, the better the model is at distinguishing given classes. The ideal value for AUC is 1.

In this study, the 2-class confusion matrix was expanded to 4 classes, as the number of our classes was 4. Presicion, Recall, F1 score values are also required to be converted to 4-class format in order to compare success rates. As indicated in Figure 4.4, the left side represents the predicted and the top represents the actual values.



Actual Values

Predicted Values

	A	B	C	D
A	TP_A	E_{AB}	E_{AC}	E_{AD}
B	E_{BA}	TPB	E_{BC}	E_{BD}
C	ECA	ECB	TPC	ECD
D	EDA	EDB	EDC	TPD

Figure 4.4 : Expanded confusion matrix with 4 classes matrix.

The aim in the 2-class confusion matrix are also valid here. The aim is to keep the diagonal variables (TPA, TPB, TPC, TPD.) at the highest level. However, the number of variables used in calculations increases from 4 to 16 in a 4-class matrix. This means that it is necessary to make changes in the calculation criteria to determine how successful the proposed models are. The calculation methods obtained from the confusion matrix in the 4-class system are as follows,

<u>Accuracy of 4 Classes System:</u> It can be expressed as the division of correctly known values to all values in the confusion matrix.

$$Accuracy = \frac{\mathrm{TP_A} + \mathrm{TP_B} + \ \mathrm{TP_C} + \ \mathrm{TP_D}}{All} \qquad (4.5)$$

<u>Precision of 4 Classes System</u>: It is a performance measurement system where False Positive values are important. Because there are multiple classes, different precision values occur for each class.

$$Precision\ A = \frac{\mathrm{TP_A}}{\mathrm{TP_A} + E_{AB} + E_{AC} + E_{AD}} \qquad (4.6)$$

$$Precision\ B = \frac{\mathrm{TP_B}}{\mathrm{TP_B} + E_{BA} + E_{BC} + E_{BD}} \tag{4.7}$$

Recall of 4 Classes System: It is a performance measurement system in which false negative values are considered, and likewise, the recall value differs for each class.

$$Recall\ A = \frac{\mathrm{TP_A}}{\mathrm{TP_A} + E_{BA} + E_{CA} + E_{DA}} \tag{4.8}$$

$$Recall\ B = \frac{\mathrm{TP_B}}{\mathrm{TP_B} + E_{AB} + E_{CB} + E_{DB}} \tag{4.9}$$

F1-Score of 4 Classes System: There must be a different F1-Score value for each class.

4.3 Number of Weights in CNN Models

One of the main reasons for creating this thesis is the number of parameters used in pre-trained models. According to the general belief, the reason for the high accuracy rate is directly proportional to the amount of parameter used. Our main goal in this study is to show that the opposite of this situation may also be true. In other words, the situation of achieving success close to pre-trained models by using much less number of parameters. The number of layers and parameters of the pre-trained models are shown in Table 4.2. In addition, the image sizes requested by the models at the entrance have been added. Parameter numbers in pre-trained models are created from "pytorch/vision:v0.10.0" in the pytorch library and may differ according to other libraries and versions. However, it is not expected that there will be a different value in the number of layers in different libraries.

Table 4.2 : Number of parameters of CNN models.

Model	Layers	Number of Parameters	Default Input Size
AlexNet	8	57,004,740	(224,224,3)
VGG-16	16	134,275,780	(224,224,3)
VGG-19	19	139,585,476	(224,224,3)
GoogleNet	22	5,597,732	(224,224,3)
MobileNet-V2	53	2,228,420	(224,224,3)
SqueezeNet	18	728,068	(224,224,3)
ResNet-34	34	21,280,452	(224,224,3)
Resnet-50	50	23,509,956	(224,224,3)
Inception-V3	42	25,119,884	(299,299,3)
Proposed Model	**6**	**264,582**	**(128,128,1)**

As seen in Table 4.2, 9 pre-trained models were determined. In these pre-trained models, except for SqueezeNet, the parameter numbers of all other models are in the millions level. In the proposed Model, this number is just 264,582 and this number is quite small when compared to other models. For example, VGG-19, which is the heaviest model among the pre-trained models, has almost 140 million parameters, which is almost 600 times the model we proposed. The number of layers and the default input size are also less than other models have.

Another important issue with models is their weight. Models with many parameters are also heavy models. It is very difficult for very heavy models to exist in mobile devices due to limited resources, and the use of applications in the field of computer vision on mobile devices is increasing day by day. For this reason, the importance of creating lightweight models is quite important.

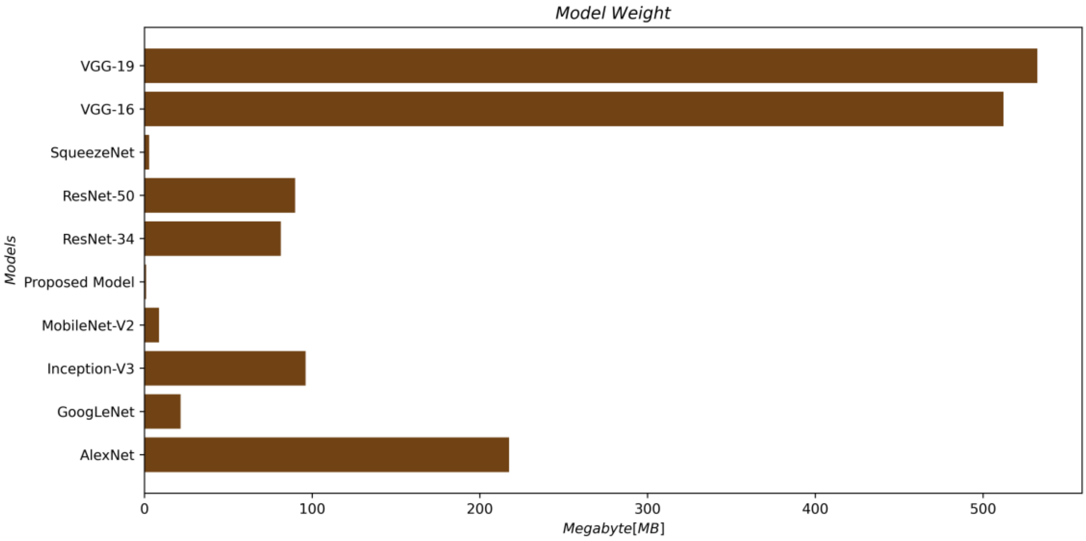


Figure 4.5 : The graph of weights of CNN models.

In Figure 4.5, it is shown the comparisons of the models in terms of their weights. Weights are also shown MB (Megabyte) level. the heaviest model in the graph is VGG-19 and is approximately 530 MB in size. Next comes VGG-16 with a weight of 510 MB. The lightest pre-trained model is SqueezeNet with 2.8 MB and the proposed model is the lightest model in the group of pre-trained models with 1 MB. Compared to VGG-19, the proposed model, which is 500 times lighter, is also advantageous in terms of low memory need.

4.4 Training Times of CNN Models

A large number of parameters also means the time that needs to be spent. So, the more parameters there are, the more variables that need to be updated in each epoch. Considering hardware constraints (Memory, GPU, Power), another issue that needs to be considered is the time spent in training time along with success. Because no matter how good the model is, if the time required for training is too much, unfortunately the heavy model is not preferred. In addition, this time will be much longer as the model will enter the training process with different hyperparameters to find the best hyperparameters

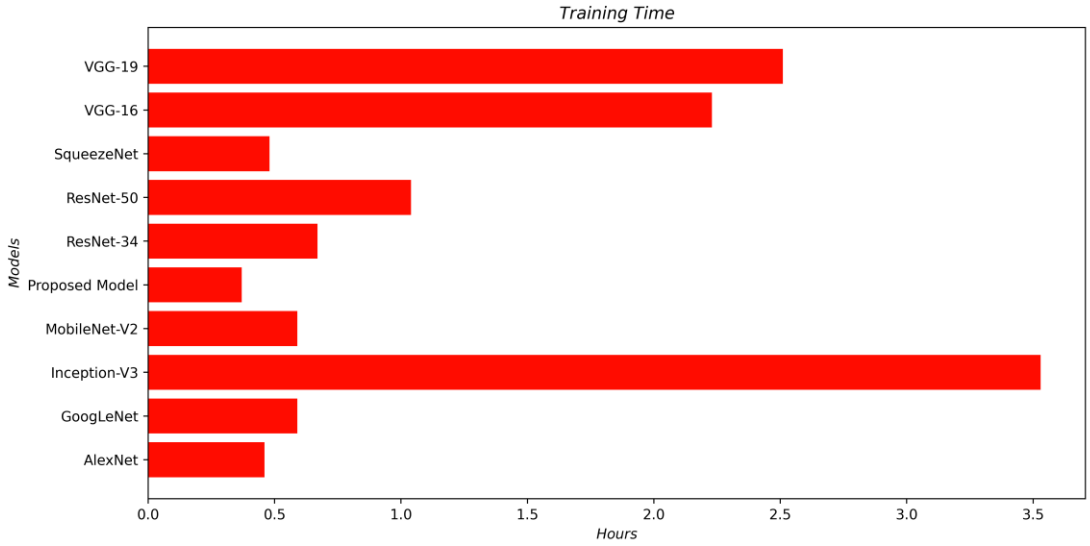


Figure 4.6 : Training times of CNN models.

In Figure 4.6, the times required to train the models are given in hours. The image input sizes have been changed to (128x128x1) except for Inception-V3. In Inception-V3, unlike other models, since it was not enough to modify the first convolution layer, images having (299x299x1) were included in the training process in order not to disturb the structure of the model. For this reason, it is estimated that it needs much more time than other models. Also, since we have 4 classes, the output layers of all models have been changed to 4 neurons.

As can be seen, the proposed model has undergone a faster training process than all other models. The training process consists of 30 epochs for all models. The closest model in terms of time spent on the proposed model is SqueezeNet. For a clear comparison, the proposed model completed 30 epochs in 1338 seconds (0.37 hr), while the Inception-V3 completed it in 12727 seconds (3.54 hr). In other words, the proposed model completed the training process approximately 10 times earlier.

4.5 Comparison of Performance Results

In this section, the results obtained and the comparisons of the proposed model with other models will be shared with various figures. In order to find out what value the proposed model is most suitable for, different learning rates were first tried and the most suitable values were selected and the comparison with the pre-trained models was started. All the graphs are plotted with the functions provided by the Matplotlib library. The reason why the epoch value was determined as 30 is because the training accuracy value of the model with the highest success in the pre-trained models is almost 100% at 30 epochs.

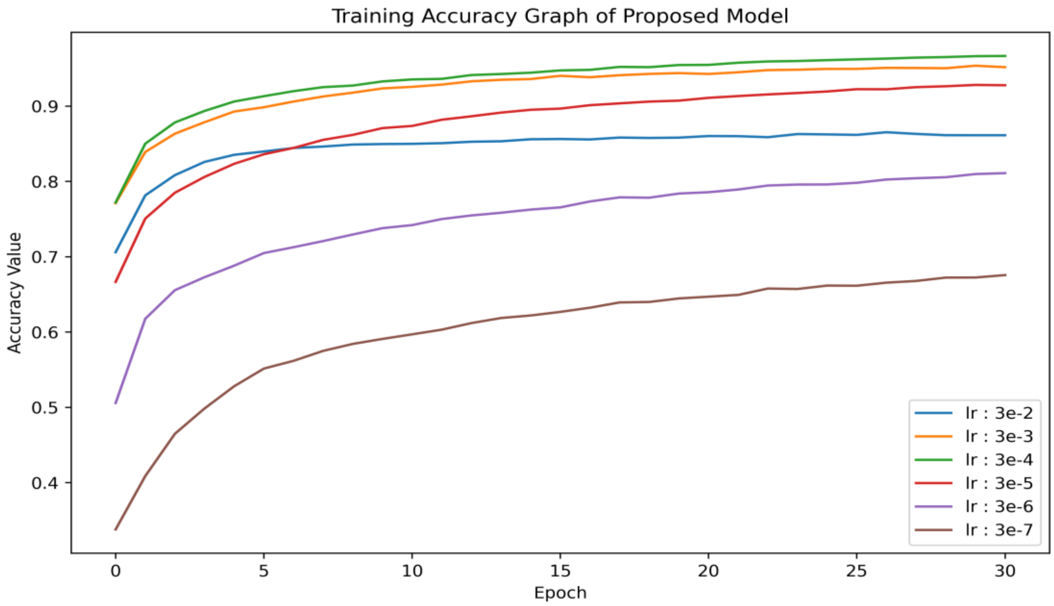


Figure 4.7 : Training accuracy graph of proposed model with diffierent learning rates.

In Figure 4.7, accuracy values of proposed model regarding different learning rates during 30 epochs are given. As can be seen, the value of the learning rate to be selected is 3e-4. Choosing a larger value causes it not to learn after a certain epoch, and choosing a smaller value causes it to learn very slowly. For this reason, the most appropriate learning rate in the training accuracy graph is seen as 3e-4. Of course, that's not the only factor. The final learning rate will be determined by examining the loss and validation graphs. In Figure 4.8, the loss values that occur during training are shared.

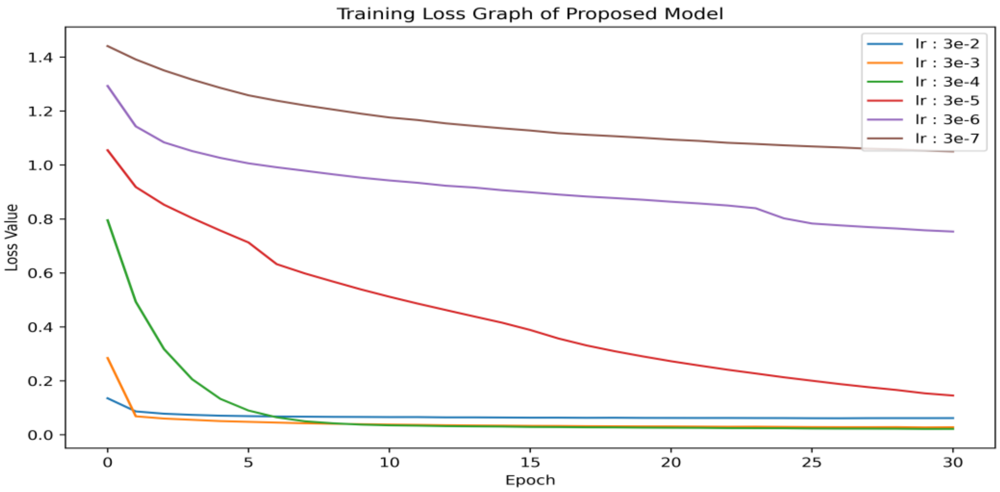


Figure 4.8 : Training loss graph of proposed model with diffierent learning rates.

The learning rates that reduce the loss value the most are 3e-4,3e-3 and 3e-2, respectively. 3e-2 and 3e-4 followed an almost constant path after the first 5 epochs. However, the 3e-4 learning rate value led a rapid loss decrease up to the first 10 epochs, followed by small decreases up to the 30th epoch.

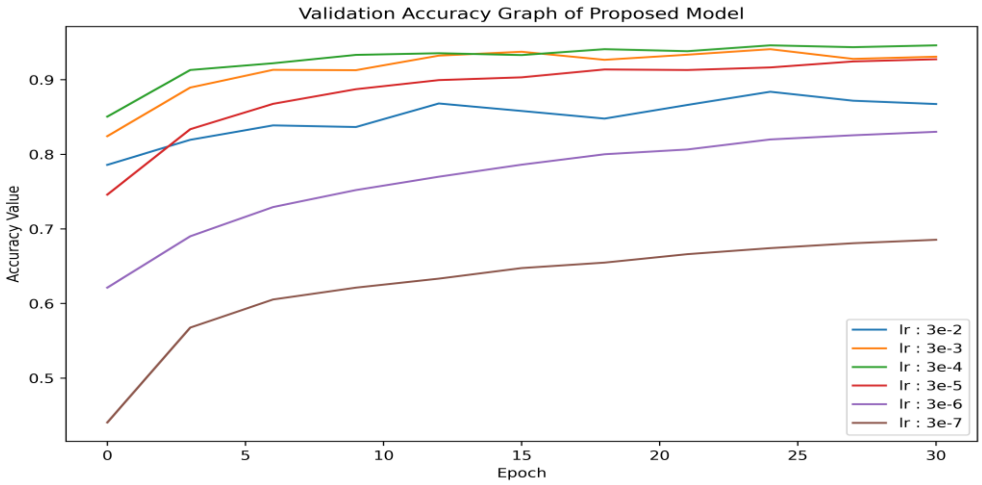


Figure 4.9 : Validation accuracy graph of proposed model with diffierent learning rates.

Accuracy and loss graphs are created on the validation set to observe whether the model is overfitted during the training. The critical point is that the model is not learning during these measurements and is faced with these images for the first time. As stated in Section 4.1, the dataset is divided into 3 different parts and only the train set images determined for training are used in the learning of the model.

The validation set provides measurement during the training, while the test set is required for final evaluation. Validation accuracy graph is shared in Figure 4.9. It is understood from the graph that the best value for accuracy is 3e-4.

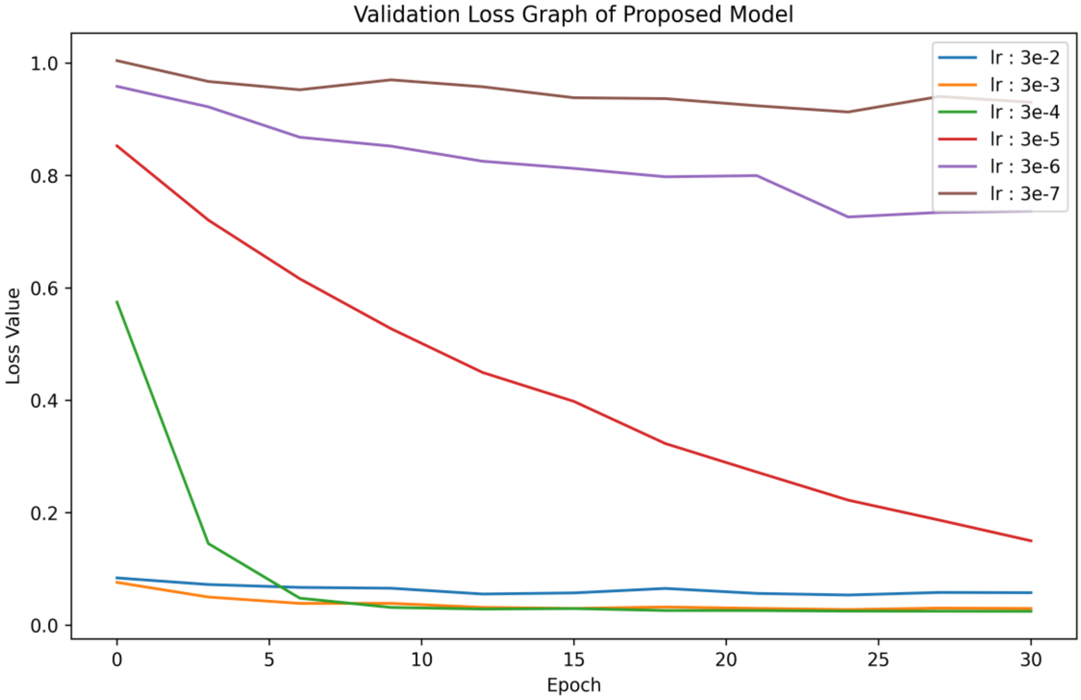


Figure 4.10 : Validation loss graph of proposed model with diffierent learning rates.

In Figure 4.10, the loss values created by the validation set in the model are seen. The learning rate that contributed the most to the decrease of loss is seen in the learning rate of 3e-4.

As a result of the 4 graphics shared above (Figure 4.7, Figure 4.8, Figure 4.9, Figure 4.10), it was observed that the most appropriate values for the learning rate were 3e-4. The values obtained by using this value will be compared in comparisons with the pretained models.

Table 4.3 : Best hyperparameters value of proposed model.

Hyperparameters	**Values**
Batch Size	64
Learning Rate	3e-4
Weight Decay	3e-6
Epoch	30
Optimizer	Adam
Number of Layer	6
Dropout Rate	0.9
Kernel Size in Convolution	5x5, (4x4 in the last layer)
Number of Kernel in Convolution	50, (16 in the last layer)
Criteria (Loss Function)	Mean Square Error
Number of Worker	12, (for test computer)

Hyperparameter is not just about learning rate. Measurements are required to determine many more values. In Table 4.3, the hyperparameter values determined by trying different combinations for the proposed model are shared. Number of Worker, which is a hyperparameter of the DataLoader offered by the Pytorch library, is used for faster reading of data. Although it does not affect the success of the model, it does affect the training time. During the comparisons with the pre-trained models, this figure was kept constant for each model and it was aimed to make the results unbiased.

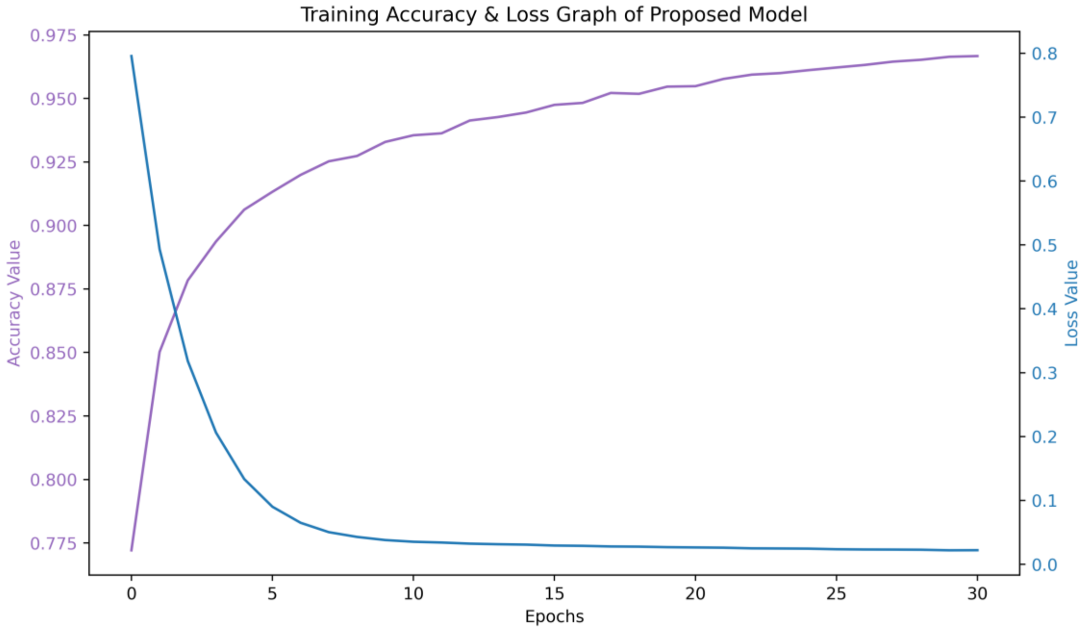


Figure 4.11 : Training accuracy & loss graph of proposed model with final hyperparameter variables.

In Figure 4.11, training accuracy and loss values created with the most appropriate hyperparameters are shared. Learning continued for 30 epochs, increasing the accuracy rate to 97%. Loss value also decreased to 0.01 level, reaching the lowest value among epochs.

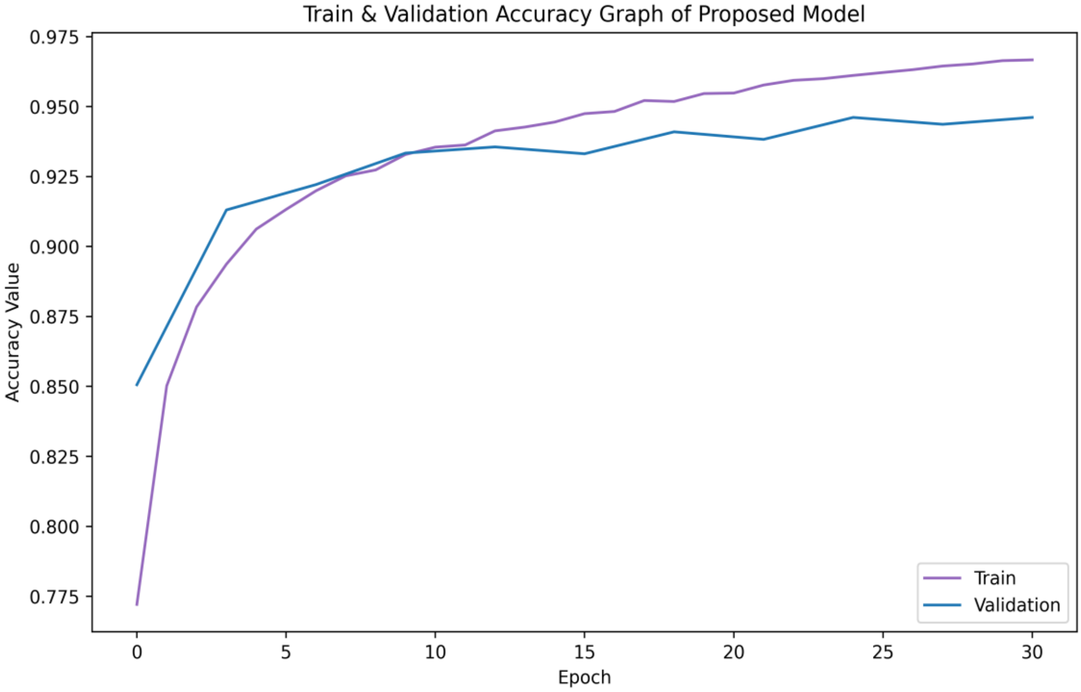


Figure 4.12 : Train and validation accuracy graph of proposed model.

Train and Validation graphs are checked to understand whether overfitting has occurred during the training. If there is no correlation in train and validation, it is suggested that the model is overfitting. So, we expect to see similar upward and downward movements in both cases.

Looking at the graph in Figure 4.12, it is seen that there is no situation that can be understood as overfitting and for this situation, many operations such as normalization and dropout were applied during model learning and it was tried to prevent the model from memorizing.

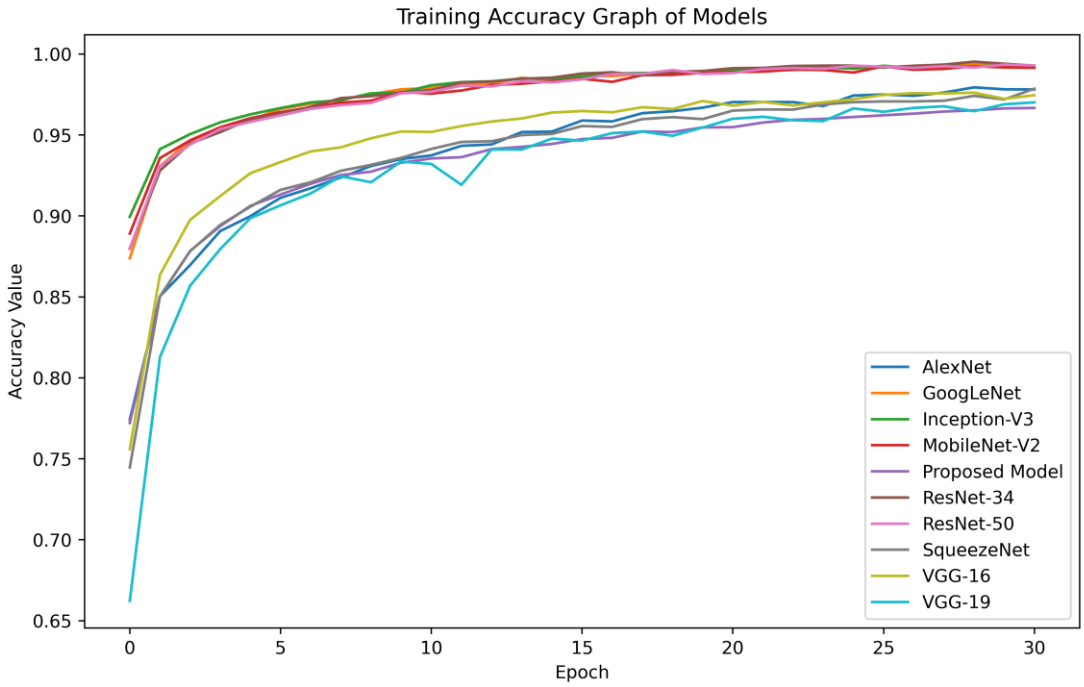


Figure 4.13 : Training accuracy graph of CNN models.

The comparison of the 9 different pre-trained and the proposed model was started with Figure 4.13. As can be seen, the proposed model is approximately 3 points behind the model with the highest accuracy, with an accuracy of 96.6%. The pre-trained model with the highest accuracy rate is Inception-V3 with 99.2%. According to the graph, the worst performing pre-trained models were observed as VGG-19.

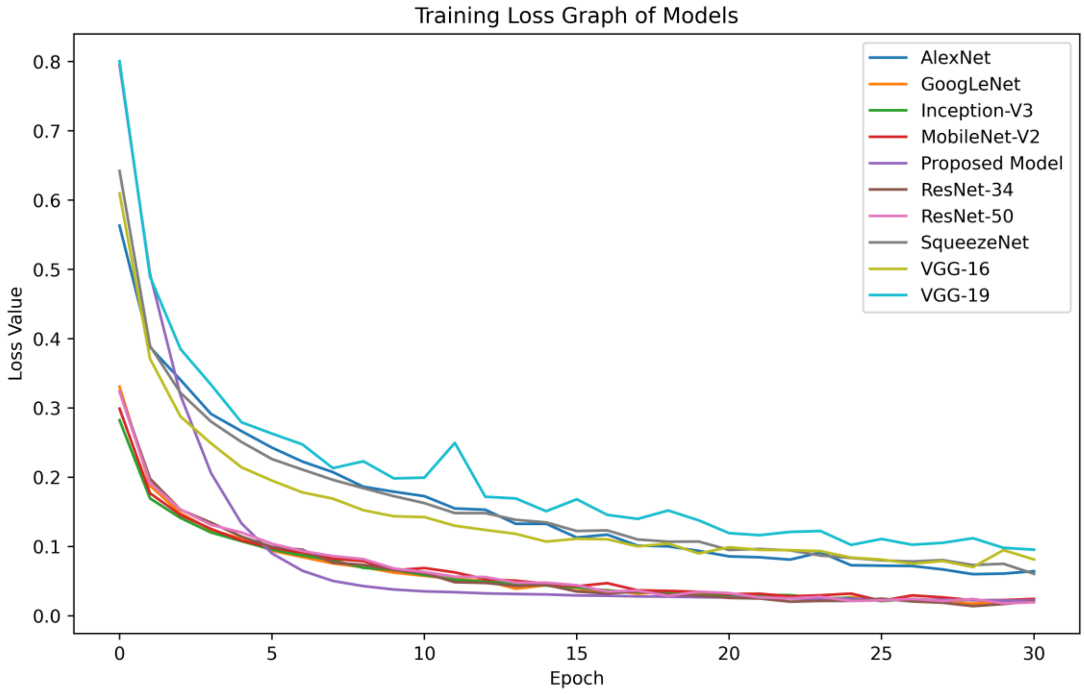


Figure 4.14 : Training loss graph of CNN models.

In Figure 4.14, it is shown the loss values created by the models during 30 epochs. It is noteworthy that VGG-19 and the proposed model start with the highest loss value.

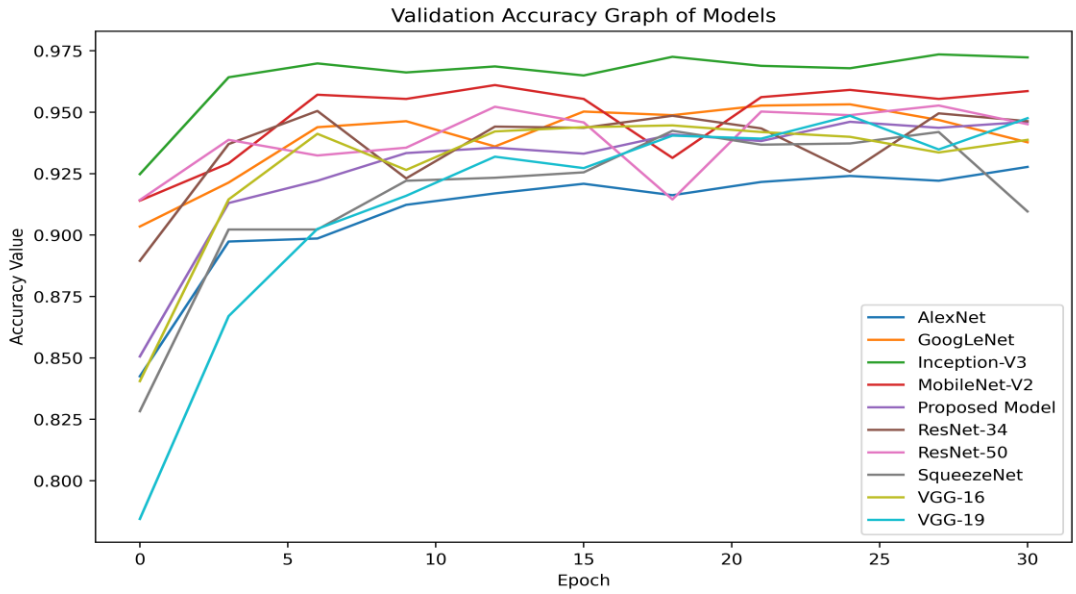


Figure 4.15 : Validation accuracy graph of CNN models.

In Figure 4.15, it is shown the validation accuracy values created by the models during 30 epochs. Inception-V3 model has the most best values across all epochs.

Contrary to Training, the proposed model performed better than most of the other models. The Validation Accuracy value was measured as 94.61% for the proposed model.

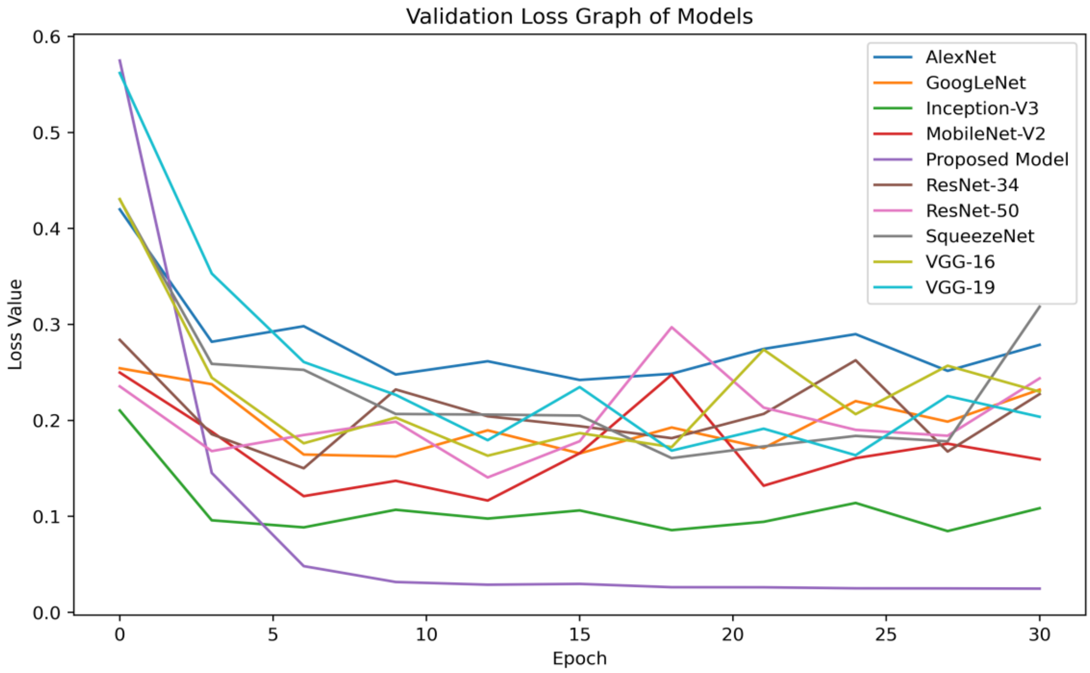


Figure 4.16 : Validation loss graph of CNN models.

In Figure 4.16, it is shown the validation loss values created by the models during 30 epochs. The propsed model reduces the loss value the most.

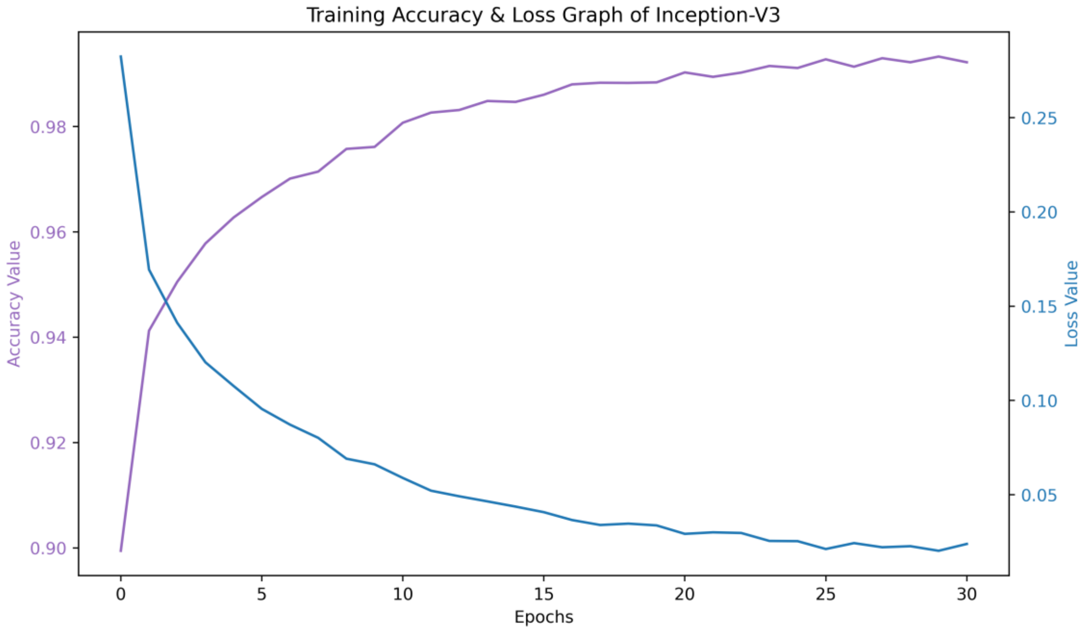


Figure 4.17 : Training accuracy & loss graph of Inception-V3.

The accuracy & loss graph of Inception-V3, which is the best performing model among the pre-trained models, is given in Figure 4.17. While the left side shows the accuracy values, the right side of the graph shows the loss values created during the epochs.

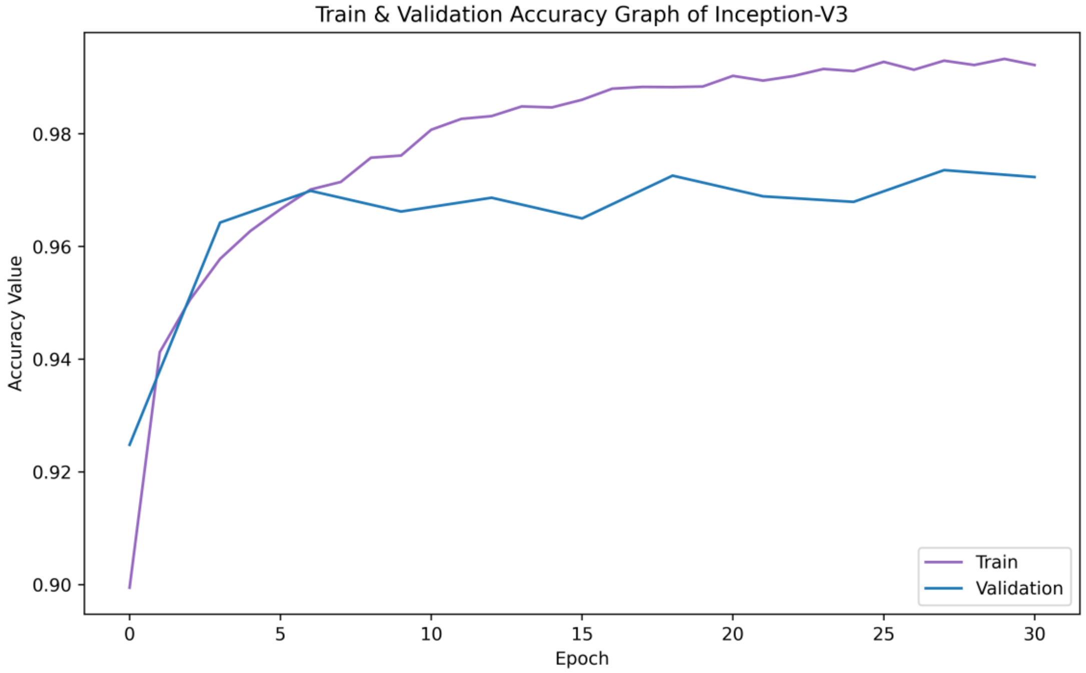


Figure 4.18 : Train & validation accuracy graph of Inception-V3.

Figure 4.18 was created to observe whether the best performing Inception-V3 is overfitting. After 10 epochs, although the difference between the train and validation accuracy values increases, this rate does not exceed 2%. In this respect, it can be concluded that there is no overfit.

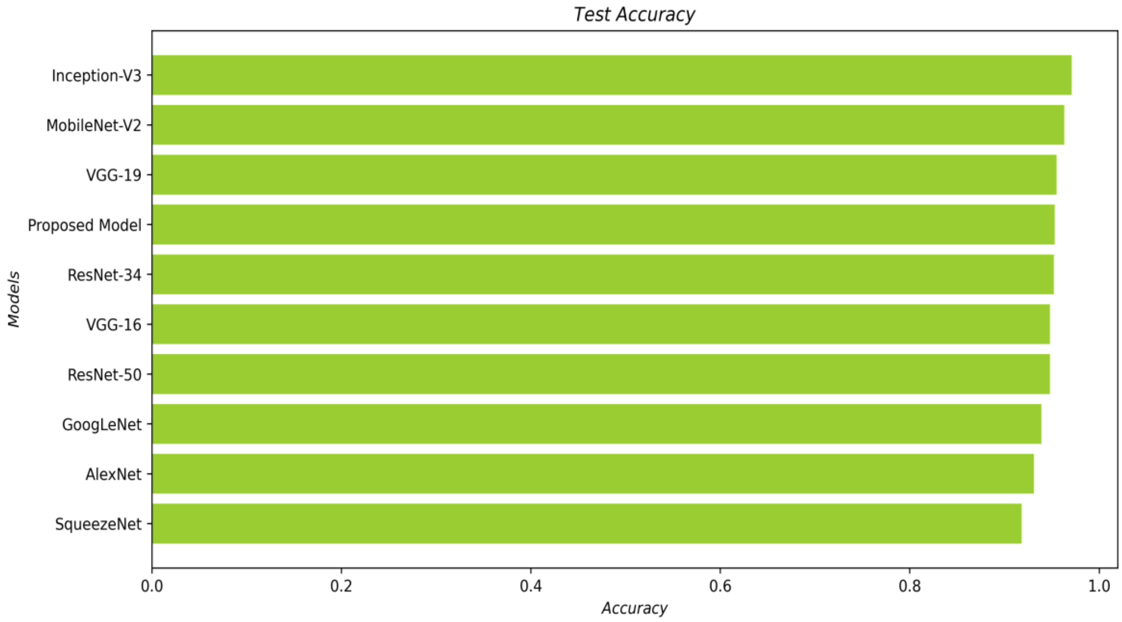


Figure 4.19 : Test accuracy graph of CNN models.

The performances of the models during training and validation so far have been examined. Now, measurements have been made about how successful they were during the test.

Figure 4.18 shows the accuracy values that the models show when the test data is given.

As can be seen, although the pre-trained models have millions of parameters, the proposed model comes in the 4th place behind Inception-v3, MobileNet-V2 and VGG-19 in the test accuracy graph with a small number of parameters.

Table 4.4 shows the confusion matrix of Inception-V3. This matrix created with test data contains 8167 images.

Table 4.4 : Confusion matrix of Inception-V3.

Class	Covid-19	Lung Opacity	Normal	Viral Pneumonia	Classification Overall	Precision
Covid-19	1997	28	17	3	2045	97.653%
Lung Opacity	9	1935	83	1	2028	95.414%
Normal	5	65	1952	17	2039	95.733%
Viral Pneumonia	1	0	1	2053	2055	99.903%
Truth Overall	2012	2028	2053	2074	8167	
Recall	99.254%	95.414%	95.08%	98.987%		97.18%
F1-Score	98%	95%	95%	99%		

Precision, Recall and F1-Score values are also shared. Inception-V3 was chosen because it has the highest accuracy value in training and validation datasets.

Table 4.5 : Confusion matrix of propsed model.

Class	Covid-19	Lung Opacity	Normal	Viral Pneumonia	Classification Overall	Precision
Covid-19	1955	42	34	14	2045	95.59%
Lung Opacity	8	1817	123	8	1956	92.89%
Normal	11	59	1962	7	2039	96.224%
Viral Pneumonia	1	0	4	2050	2055	99.757%
Truth Overall	1975	1918	2123	2079	8095	
Recall	98.98%	94.73%	92.416%	98.60%		96.15%
F1-Score	97%	94%	94%	99%		

Table 4.5 shows the confusion matrix of the proposed model. This matrix created with test data contains 8095 images. Precision, Recall and F1-Score values are also shared.

5. CONCLUSIONS AND DISCUSSION

With Covid-19, the importance of chest radiographs in our lives is increasing day by day. The X-Ray method, which is frequently used for diagnosis and pre-diagnosis, provides great convenience as it does not require a physical operation. It is accepted as a safer method with its relatively low radiation hazard. It is seen as a fast and inexpensive choice compared to other methods (MRI, CT, ultrasound, etc.) thanks to its imaging with X-Ray rays. Chest X-rays taken for imaging the diseases in the lung and chest region are the source of this study. It is aimed to support these images, which require expertise and time to read and interpret, with computer technologies in order to read them faster and more accurately. In this direction, artificial intelligence techniques, which have been very popular recently, have been used and various functions have been proposed in order to differentiate and improve existing models.

In this study, it is aimed to classify chest radiographs with 4 different conditions. Images taken from public sources have been augmented and the number of images between classes has been equalized. In this way, it is ensured that there is no dominant class that will affect the decision of the model. The prepared data set is divided for train, test and validation processes as 70%, 20% and 10%.

In order to evaluate the performance, pre-trained models (AlexNet, VGG-16, VGG-19, GoogleNet, MobileNet-V2, SqueezeNet, ResNet-34, Resnet-50, Inception-V3) were trained and their success rates, time to be spent, required parameter numbers and hyperparameters to be calculated (number of nodes in the hidden layer) were compared with the proposed method.

As a result, it was observed that the proposed model made better predictions than many pre-trained models against the test data and ranked 4th with an accuracy rate of 96.15%. Comparing the training time, the proposed model completed it’s training in the shortest time among all models and it also showed that it is the lightest model in the model weight comparison. Thus, it is thought to be a very useful model in reducing the processing load, maintaining the accuracy rate, and reducing the need for hyperparameters.

REFERENCES

[1] **K. Simonyan and A. Zisserman**, “VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION,” 2015, Accessed: Jan. 03, 2022. [Online]. Available: http://www.robots.ox.ac.uk/

[2] **J. P. Mizgerd**, “Inflammation and Pneumonia: Why Are Some More Susceptible than Others?,” *Clin Chest Med*, vol. 39, no. 4, p. 669, Dec. 2018, doi: 10.1016/J.CCM.2018.07.002.

[3] **R. Jain, M. Gupta**, S. Taneja, and D. J. Hemanth, “Deep learning based detection and analysis of COVID-19 on chest X-ray images,” *Applied Intelligence*, vol. 51, no. 3, pp. 1690–1700, Mar. 2021, doi: 10.1007/S10489-020-01902-1/FIGURES/10.

[4] **C. Szegedy, V. Vanhoucke, S. Ioffe**, J. Shlens, and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-December, pp. 2818–2826, Dec. 2016, doi: 10.1109/CVPR.2016.308.

[5] **F. Chollet**, “Xception: Deep learning with depthwise separable convolutions,” *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 1800–1807, Nov. 2017, doi: 10.1109/CVPR.2017.195.

[6] **S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He**, “Aggregated Residual Transformations for Deep Neural Networks,” *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 5987–5995, Nov. 2016, doi: 10.1109/CVPR.2017.634.

[7] **S. Karakanis and G. Leontidis**, “Lightweight deep learning models for detecting COVID-19 from chest X-ray images,” *Computers in Biology and Medicine*, vol. 130, p. 104181, Mar. 2021, doi: 10.1016/J.COMPBIOMED.2020.104181.

[8] **C. Ouchicha, O. Ammor, and M. Meknassi**, “CVDNet: A novel deep learning architecture for detection of coronavirus (Covid-19) from chest x-ray images,” *Chaos, Solitons, and Fractals*, vol. 140, p. 110245, Nov. 2020, doi: 10.1016/J.CHAOS.2020.110245.

[9] **S. Basu, S. Mitra, and N. Saha**, “Deep Learning for Screening COVID-19 using Chest X-Ray Images,” *2020 IEEE Symposium Series on Computational*

Intelligence, SSCI 2020, pp. 2521–2527, Apr. 2020, doi: 10.1109/SSCI47803.2020.9308571.

[10] **A. Krizhevsky, I. Sutskever, and G. E. Hinton**, "ImageNet Classification with Deep Convolutional Neural Networks", Accessed: Jan. 03, 2022. [Online]. Available: http://code.google.com/p/cuda-convnet/

[11] **K. Simonyan and A. Zisserman**, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, Sep. 2014, Accessed: Jan. 03, 2022. [Online]. Available: https://arxiv.org/abs/1409.1556v6

[12] **K. He, X. Zhang, S. Ren, and J. Sun**, "Deep residual learning for image recognition," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-December, pp. 770–778, Dec. 2016, doi: 10.1109/CVPR.2016.90.

[13] **K. Shankar and E. Perumal**, "A novel hand-crafted with deep learning features based fusion model for COVID-19 diagnosis and classification using chest X-ray images," *Complex & Intelligent Systems 2020 7:3*, vol. 7, no. 3, pp. 1277–1293, Nov. 2020, doi: 10.1007/S40747-020-00216-6.

[14] **P. Bhowal, S. Sen, J. H. Yoon, Z. W. Geem, and R. Sarkar**, "Choquet Integral and Coalition Game-Based Ensemble of Deep Learning Models for COVID-19 Screening From Chest X-Ray Images," *IEEE J Biomed Health Inform*, vol. 25, no. 12, pp. 4328–4339, Dec. 2021, doi: 10.1109/JBHI.2021.3111415.

[15] **E. Hussain, M. Hasan, M. A. Rahman, I. Lee, T. Tamanna, and M. Z. Parvez**, "CoroDet: A deep learning based classification for COVID-19 detection using chest X-ray images," *Chaos, Solitons & Fractals*, vol. 142, p. 110495, Jan. 2021, doi: 10.1016/J.CHAOS.2020.110495.

[16] **A. U. Ibrahim, M. Ozsoz, S. Serte, F. Al-Turjman, and P. S. Yakoi**, "Pneumonia Classification Using Deep Learning from Chest X-ray Images During COVID-19," *Cognitive Computation*, vol. 1, pp. 1–13, Jan. 2021, doi: 10.1007/S12559-020-09787-5/TABLES/9.

[17] **S. Sakib, T. Tazrin, M. M. Fouda, Z. M. Fadlullah, and M. Guizani**, "DL-CRC: Deep learning-based chest radiograph classification for covid-19 detection: A novel approach," *IEEE Access*, vol. 8, pp. 171575–171589, 2020, doi: 10.1109/ACCESS.2020.3025010.

[18] **Nur-a-alam, M. Ahsan, M. A. Based, J. Haider, and M. Kowalski**, "COVID-19 Detection from Chest X-ray Images Using Feature Fusion and Deep Learning," *Sensors 2021, Vol. 21, Page 1480*, vol. 21, no. 4, p. 1480, Feb. 2021, doi: 10.3390/S21041480.

[19] **S. R. Nayak, D. R. Nayak, U. Sinha, V. Arora, and R. B. Pachori**, "Application of deep learning techniques for detection of COVID-19 cases using chest X-ray images: A comprehensive study," *Biomedical Signal*

Processing and Control, vol. 64, p. 102365, Feb. 2021, doi: 10.1016/J.BSPC.2020.102365.

[20] **C. Szegedy *et al.*, "Going deeper with convolutions**," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 07-12-June-2015, pp. 1–9, Oct. 2015, doi: 10.1109/CVPR.2015.7298594.

[21] **M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen**, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 4510–4520, Dec. 2018, doi: 10.1109/CVPR.2018.00474.

[22] **F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer**, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size," Feb. 2016, Accessed: Jan. 03, 2022. [Online]. Available: https://arxiv.org/abs/1602.07360v4

[23] **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," 2020, doi: 10.1038/s41598-020-76550-z.

[24] "**X-Ray Waves - Waves and Health**." https://sites.google.com/site/wavesandhealth/x-ray-waves (accessed Mar. 14, 2022).

[25] **W. Pezzotti**, "Chest X-ray interpretation: Not just black and white," *Nursing (Brux)*, vol. 44, no. 1, pp. 40–47, Jan. 2014, doi: 10.1097/01.NURSE.0000438704.82227.44.

[26] "**Chest x-ray: MedlinePlus Medical Encyclopedia**." https://medlineplus.gov/ency/article/003804.htm (accessed Jan. 04, 2022).

[27] "**Chest X-Ray | NHLBI, NIH**." https://www.nhlbi.nih.gov/health-topics/chest-x-ray (accessed Jan. 04, 2022).

[28] "**Interpreting A Chest X-Ray - Stepwards**." https://www.stepwards.com/?page_id=4353 (accessed Mar. 14, 2022).

[29] "**X-ray (Chest)**." http://imagingpathways.health.wa.gov.au/index.php/consumer-info/imaging-procedures/chest-x-ray (accessed Mar. 14, 2022).

[30] "**Portable chest X-ray in coronavirus disease-19 (COVID-19)_ A pictorial review | Elsevier Enhanced Reader**." https://reader.elsevier.com/reader/sd/pii/S0899707120301017?token=46C040CD4654D2B7F0D67B7A0F61C73F1B872B97C3390EEB147198018AB5BBE62B98C37B7C31372E2B037D4E450D20BF&originRegion=eu-west-1&originCreation=20220104193733 (accessed Jan. 04, 2022).

[31] **H. Y. F. Wong *et al.***, "Frequency and Distribution of Chest Radiographic Findings in Patients Positive for COVID-19," *Radiology*, vol. 296, no. 2, pp.

E72–E78, Aug. 2020, doi: 10.1148/RADIOL.2020201160/ASSET/IMAGES/LARGE/RADIOL.2020201160.FIG6.JPEG.

[32] **J. E. Bourcier *et al*.**, “Performance comparison of lung ultrasound and chest x-ray for the diagnosis of pneumonia in the ED,” *The American Journal of Emergency Medicine*, vol. 32, no. 2, pp. 115–118, Feb. 2014, doi: 10.1016/J.AJEM.2013.10.003.

[33] “**WHO Coronavirus (COVID-19) Dashboard** | WHO Coronavirus (COVID-19) Dashboard With Vaccination Data.” https://covid19.who.int/ (accessed Jan. 01, 2022).

[34] **X. He, W. Hong, X. Pan, G. Lu, and X. Wei**, “SARS-CoV-2 Omicron variant: Characteristics and prevention,” *MedComm (Beijing)*, vol. 2, no. 4, pp. 838–845, Dec. 2021, doi: 10.1002/MCO2.110.

[35] **J. A. Plante, B. M. Mitchell, K. S. Plante, K. Debbink, S. C. Weaver, and V. D. Menachery**, “The variant gambit: COVID-19’s next move,” *Cell Host & Microbe*, vol. 29, no. 4, pp. 508–515, Apr. 2021, doi: 10.1016/J.CHOM.2021.02.020.

[36] “**New Images of Novel Coronavirus SARS-CoV-2 Now Available | NIH: National Institute of Allergy and Infectious Diseases**.” https://www.niaid.nih.gov/news-events/novel-coronavirus-sarscov2-images (accessed Mar. 14, 2022).

[37] **T. Stubbs, W. Kring, C. Laskaridis, A. Kentikelenis, and K. Gallagher**, “Whatever it takes? The global financial safety net, Covid-19, and developing countries,” *World Development*, vol. 137, p. 105171, Jan. 2021, doi: 10.1016/J.WORLDDEV.2020.105171.

[38] **T. Ahmad *et al*.**, “COVID-19: Zoonotic aspects,” *Travel Medicine and Infectious Disease*, vol. 36, p. 101607, Jul. 2020, doi: 10.1016/J.TMAID.2020.101607.

[39] **P. Zhou *et al*.**, “A pneumonia outbreak associated with a new coronavirus of probable bat origin,” *Nature*, vol. 579, no. 7798, pp. 270–273, Mar. 2020, doi: 10.1038/S41586-020-2012-7.

[40] **C. Biscayart, P. Angeleri, S. Lloveras, T. do S. S. Chaves, P. Schlagenhauf, and A. J. Rodríguez-Morales**, “The next big threat to global health? 2019 novel coronavirus (2019-nCoV): What advice can we give to travellers? – Interim recommendations January 2020, from the Latin-American society for Travel Medicine (SLAMVI),” *Travel Medicine and Infectious Disease*, vol. 33, p. 101567, Jan. 2020, doi: 10.1016/J.TMAID.2020.101567.

[41] **D. Wang *et al*.**, “Clinical Characteristics of 138 Hospitalized Patients With 2019 Novel Coronavirus–Infected Pneumonia in Wuhan, China,” *JAMA*, vol. 323, no. 11, pp. 1061–1069, Mar. 2020, doi: 10.1001/JAMA.2020.1585.

[42] **M. Ciotti, M. Ciccozzi, A. Terrinoni, W. C. Jiang, C. bin Wang, and S. Bernardini**, “The COVID-19 pandemic,” *https://doi.org/10.1080/10408363.2020.1783198*, vol. 57, no. 6, pp. 365–388, 2020, doi: 10.1080/10408363.2020.1783198.

[43] **T. P. Velavan and C. G. Meyer**, “The COVID-19 epidemic,” *Tropical Medicine & International Health*, vol. 25, no. 3, p. 278, Mar. 2020, doi: 10.1111/TMI.13383.

[44] **W. Guan *et al.***, “Clinical characteristics of 2019 novel coronavirus infection in China,” *medRxiv*, p. 2020.02.06.20020974, Feb. 2020, doi: 10.1101/2020.02.06.20020974.

[45] **B. Pfefferbaum and C. S. North**, “Mental Health and the Covid-19 Pandemic,” *New England Journal of Medicine*, vol. 383, no. 6, pp. 510–512, Aug. 2020, doi: 10.1056/NEJMP2008017/SUPPL_FILE/NEJMP2008017_DISCLOSURES.PDF.

[46] **C. Wang *et al.*,** “Immediate Psychological Responses and Associated Factors during the Initial Stage of the 2019 Coronavirus Disease (COVID-19) Epidemic among the General Population in China,” *International Journal of Environmental Research and Public Health 2020, Vol. 17, Page 1729*, vol. 17, no. 5, p. 1729, Mar. 2020, doi: 10.3390/IJERPH17051729.

[47] **S. J. Daniel**, “Education and the COVID-19 pandemic,” *PROSPECTS 2020 49:1*, vol. 49, no. 1, pp. 91–96, Apr. 2020, doi: 10.1007/S11125-020-09464-3.

[48] **M. J. Delahoy *et al.***, “Hospitalizations Associated with COVID-19 Among Children and Adolescents — COVID-NET, 14 States, March 1, 2020–August 14, 2021,” *Morbidity and Mortality Weekly Report*, vol. 70, no. 36, p. 1255, 2021, doi: 10.15585/MMWR.MM7036E2.

[49] “**Infection prevention and control during health care when novel coronavirus (nCoV) infection is suspected**: interim guidance, 25 January 2020.” https://apps.who.int/iris/handle/10665/330674 (accessed Jan. 01, 2022).

[50] “**COVID19 Vaccine Tracker**.” https://covid19.trackvaccines.org/ (accessed Jan. 01, 2022).

[51] **T. P. Mashamba-Thompson and E. D. Crayton**, “Blockchain and Artificial Intelligence Technology for Novel Coronavirus Disease 2019 Self-Testing,” *Diagnostics 2020, Vol. 10, Page 198*, vol. 10, no. 4, p. 198, Apr. 2020, doi: 10.3390/DIAGNOSTICS10040198.

[52] **B. McCall**, “COVID-19 and artificial intelligence: protecting health-care workers and curbing the spread,” *The Lancet Digital Health*, vol. 2, no. 4, pp. e166–e167, Apr. 2020, doi: 10.1016/S2589-7500(20)30054-6.

[53] **Ö. Polat, Z. Dokur, and T. Ölmez**, “Turk J Elec Eng & Determination of Pneumonia in X-ray Chest Images by Using Convolutional Neural Network”, doi: 10.3906/elk-2009-1.

[54] **V. K. Garg**, “Wide-Area Wireless Networks — cdmaOne Evolution,” *Wireless Communications & Networking*, pp. 541–593, 2007, doi: 10.1016/B978-012373580-5/50050-8.

[55] **R. B. Lackey, R. B. Lackey, and D. Meltzer**, “A Simplified Definition of Walsh Functions,” *IEEE Transactions on Computers*, vol. C–20, no. 2, pp. 211–213, 1971, doi: 10.1109/T-C.1971.223214.

[56] “**Walsh matrix - Wikipedia**.” https://en.wikipedia.org/wiki/Walsh_matrix (accessed Mar. 14, 2022).

[57] “**Walsh Functions - an overview** | ScienceDirect Topics.” https://www.sciencedirect.com/topics/engineering/walsh-functions (accessed Mar. 14, 2022).

[58] “**COVID-19 Radiography Database** | Kaggle.” https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database (accessed May 20, 2022).

CURRICULUM VITAE

Name Surname : Muhammed Nur Talha Kılıç

EDUCATION :

- **B.Sc. :** 2018, Selcuk University, Faculty of Engineering, Electrical and Electronics Engineering
- **M.Sc. :** 2022, Istanbul Technical University, Graduate School of Science Engineering and Technology , Electrical and Communication Engineering

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- **Kılıç, Muhammed Nur Talha, and Tamer Ölmez,** 2021. Detection of Covid-19 in Chest X-ray Image by Using Convolutional Network Trained with Walsh Functions, World Conference on Innovation in Technology and Engineering Sciences, 3-5 Dec 2021, Athens/GREECE,