Optimal Transport Theory based GAN for Medical Image Augmentation and Classification

by

Md. Abdul Kahhar Siddiki Shan
18301221
Md. Abdul Quaiyum
18301195
Sugata Saha
18301089
S. M. Navin Nayer Anik
18301189

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
School of Data and Sciences
Brac University
January 2023

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Md. Abdul Kahhar Siddiki Shan	Md. Abdul Quaiyum
18301221	18301195
Sugata Saha	S. M. Navin Nayer Anik
18301089	18301189

Approval

The thesis titled "Optimal-Transport Theory based GAN for Medical Image Augmentation and Classification" submitted by

1. Md. Abdul Kahhar Siddiki Shan (18301221)
2. Md. Abdul Quaiyum (18301195)
3. Sugata Saha (18301089)
4. S. M. Navin Nayer Anik (18301189)

Of Fall, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 19, 2023.

Examining Committee:

Supervisor:
(Member)

Dr. Md. Golam Rabiul Alam, PhD

Professor
Department of Computer Science and Engineering
Brac University

Thesis Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam, PhD

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD

Associate Professor
Department of Computer Science and Engineering
Brac University

Ethics Statement

It is imperative that the thesis be written strictly in compliance with the university's rules and regulations, as well as ethical principles for doing research. Original data has been incorporated into our thesis. We double-checked our citations and references. Each of the paper's four co-authors accepts responsibility for any violations of the thesis rule. There were several questionnaire-free tools, articles, and YouTube videos that helped us address problems. Furthermore, we would like to take this opportunity to express our gratitude to everyone who has helped us along the way. Our thesis was completed without the use of unethical tactics. The Brac University's code of ethics guides our activities.

Abstract

Unsupervised neural networks called Generative Adversarial Networks (GANs) are used to provide realistic pictures without relying on the original data. Due to disparities in illness occurrence, the challenge of class imbalance is prevalent in the medical area, making the development of these applications tough. In order to solve the issue of class imbalance, several GAN-based approaches have been presented. However, since extracting significant information from just a few pixels is laborious, these designs have not effectively trained for small-scale disorders. To address the difficulty of insufficient quantities and imbalances of medical imaging data, data augmentation methods such as image modifications have been utilized to minimize performance degradation. Optimal Transport (OT) theory allows for the development of a geometry for a space of functions, providing a definition of distance in this space, a technique for interpolating between various functions, and, in a more general sense, establishing the barycenter of a family of weighted functions. Consequently, optimal transport is presented as a fundamental tool in many applied domains. Measures with non-overlapping supports are well-suited to occupational therapy. However, using OT to train generative machines raises challenges like (1) a lack of smoothness, (2) the computational burden of evaluating OT losses, and (3) the difficulty of estimating gradients in high dimensions. Because of this, Skinhorn divergence will be utilized to create a loss family that includes both Wasserstein (OT) and Maximum Mean Discrepancy (MMD) losses. It will help us identify a sweet spot by utilizing OT's geometry and generating high-dimensional spaces from low-dimensional manifolds. Applications of the transport map for image augmentation and classification include computing geodesics between pictures and transferring one image's attributes to another. The nature of the seen items must be preserved in this situation in order to create pictures that are both physically and aesthetically convincing. Our proposed Optimal Transport Theory-based GAN model is able to detect and differentiate between chest X-ray images with a high degree of accuracy. The proposed method of image augmentation was used for the classification of normal, COVID-19-effected, lung opacity, and viral pneumonia images. In terms of accuracy, precision, recall, and f1 score, which are all tested in this paper, our proposed Optimal Transport Theory-based GAN model performs comparably on the available datasets. The suggested model is accurate to 96.94%, which is greater than any of the experimental results of CNN models utilized in this research due to the fact that the recommended model was developed.

Keywords: Optimal Transport(OT) Theory; Generative Adversarial Networks (GANs); Unsupervised Neural Networks; Wasserstein Distance; Maximum Mean Discrepancy(MMD); Chest X-Ray; Covid-19; Viral Pneumonia; Lung Opacity

Dedication

This paper is dedicated to our families and fellow team members. The constant support of the family members and the determination of the team members played a great role in making this paper a success. We could not have finished our thesis without the help of our esteemed supervisor, who has been a continual source of guidance and advice. We also dedicate this paper to him.

Acknowledgement

In the first place, we want to express our gratitude to Allah, the Almighty, for providing us with the resources, opportunities, and guidance necessary to complete this research on schedule. As a second tribute, we would like to thank our thesis supervisor, Dr. Md. Golam Rabiul Alam, for his tireless support and guidance as we tackled a difficult topic. We got through the challenges with his continuous support and comments. He made time for us and provided valuable insights to help us enhance our job. For that, we will be eternally thankful. Final words of thanks go out to the entire faculty and student body for establishing such a welcoming learning environment in which we were able to develop professionally while also doing this research to the best of our abilities.

Table of Contents

Declaration i

Approval ii

Ethics Statement iii

Abstract iv

Dedication v

Acknowledgment vi

Table of Contents vii

List of Figures ix

List of Tables xi

Nomenclature xiii

1 Introduction 1
- 1.1 Overview . . . 1
- 1.2 Research Problem . . . 3
- 1.3 Research Goals . . . 5

2 Literature Review 6
- 2.1 Background Study . . . 6
  - 2.1.1 Optimal Transport Theory (OT) . . . 6
  - 2.1.2 Generative Adversarial Networks(GANs) . . . 7
  - 2.1.3 Image Augmentation . . . 8
  - 2.1.4 Image Classification . . . 8
- 2.2 Related Works . . . 8

3 Methodology 13
- 3.1 Dataset . . . 14
  - 3.1.1 Chest X-ray Data Set . . . 14
  - 3.1.2 Lung Opacity Overview Data Set . . . 14
- 3.2 Image Pre-processing . . . 14
- 3.3 Training / Testing Data Formulation . . . 15
- 3.4 Image Classification with Transfer Learning Models . . . 15

3.4.1 VGG19 16
3.4.2 Inception V3 17
3.4.3 Xception 17
3.5 Decision Modelling 21
3.5.1 Designing Generator 21
3.5.2 Designing Discriminator 22
3.5.3 Wasserstein Distance 23
3.5.4 Wasserstein Distance based Loss Function 25
3.5.5 Sinkhorn Divergence 27

4 Performance Evaluation and Result Analysis **28**
4.1 Performance Evaluation Matrics 28
4.2 Experimental Result 30
4.3 Model Explainability through XAI 40

5 Conclusion **43**
5.1 Challenges and Future Directions 43

Bibliography **47**

List of Figures

3.1 The top level overview of the proposed method 13
3.2 Architecture of VGG19 17
3.3 VGG19 model summary 17
3.4 Architecture of Inception V3 18
3.5 Inception V3 model summary 18
3.6 Architecture of Xception. 20
3.7 Xception model summary 21
3.8 The architecture of generator neural network 22
3.9 The architecture of the discriminator neural network. 23
3.10 Wasserstein Distance Algorithm. 25

4.1 Precision, Recall, F1 Score and Accuracy bar chart comparison for the VGG19 model 30
4.2 Loss Function Graph of the VGG19 Model without OT-GAN 31
4.3 Loss Function Graph of the VGG19 Model with OT-GAN 31
4.4 Accuracy Graph of the VGG19 Model without OT-GAN 32
4.5 Accuracy Graph of the VGG19 Model with OT-GAN 32
4.6 ROC Curve for the VGG19 Model 33
4.7 Confusion matrix of the VGG19 Model 33
4.8 Precision, Recall, F1 Score and Accuracy bar chart comparison for the Inception V3 model 34
4.9 Loss Function Graph of the Inception V3 Model without OT-GAN . . 34
4.10 Loss Function Graph of the Inception V3 Model with OT-GAN . . . 35
4.11 Accuracy Graph of the Inception V3 Model without OT-GAN 35
4.12 Accuracy Graph of the Inception V3 Model with OT-GAN 36
4.13 ROC Curve for the Inception V3 Model 36
4.14 Confusion Matrix for the Inception V3 Model 37
4.15 Precision, Recall, F1 Score and Accuracy bar chart comparison for the Xception model 37
4.16 Loss Function Graph of the Xception Model without OT-GAN 38
4.17 Loss Function Graph of the Xception Model with OT-GAN 38
4.18 Accuracy Graph of the Xception Model without OT-GAN 39
4.19 Accuracy Graph of the Xception Model with OT-GAN 39
4.20 ROC Curve for the Xception Model 40
4.21 Confusion Matrix for the Xception Model 40
4.22 Heat Map images form GradCAM Analysis at the last layer of the CNN models 41

4.23 Superimposed images of the GradCAM Analysis at the last layer of the CNN models 42

List of Tables

4.1 Performance measures for the three models 30

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

ϵ Epsilon

γ Gamma

μ Mu

ν Nu

ω Omega

Π Big Pi

π Pi

σ Sigma

$\sum$ Sum

θ Theta

υ Upsilon

$ACGANs$ Auxiliary Classifier Generative Adversarial Networks

AI Artificial Intelligence

$ANT-GANs$ Abnormal to Normal Translation Generative Adversarial Networks

CNN Convolutional Neural Networks

CXR Chest X-Ray

$DCNNs$ Deep Convolutional Neural Networks

DL Deep Learning

ECG Electrocardiogram

EEG Electroencephalograms

$GANs$ Generative Adversarial Networks

$LIDC$ Lung Image Database Consortium

ML Machine Learning

OT Optimal Transport

$OT-GANs$ Optimal Transport theory based Generative adversarial networks

SWF Sliced Wasserstein Flows

$VAEs$ Variational Auto-Encoders

VGG Visual Geometry Group

$WGANs$ Wasserstein Generative Adversarial Networks

WHO World Health Organization

XAI Explainable Artificial Intelligence

Chapter 1

Introduction

1.1 Overview

The recent novel coronavirus diseases (COVID-19) outbreak, caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2), play a critical role in disease pathogenesis and clinical manifestation. This infection activates antiviral immune responses and cause uncontrolled inflammatory responses. The first instance was discovered in December 2019 in Wuhan, China. The sickness' quick worldwide spread and subsequent COVID-19 pandemic were caused by it. Inhaling air polluted by droplets, aerosols, and other tiny airborne particles carrying COVID-19 is the major way the virus is spread. As they breathe, talk, cough, sneeze, or sing, infected persons expel such particles. 6,676,645 fatalities have been reported to the WHO out of 657,430 confirmed cases of COVID-19 to date [35].

A radiological phrase used to describe hazy gray patches on CT or X-ray imaging is "ground-glass opacity". It suggests that these locations are becoming more dense. On a CT scan or X-ray, the lungs often show as being black [36]. This demonstrates that they are unblocked. Instead, if gray regions are observed, something is partially occupying this space within the lungs. Ground-glass opacity is the term used to describe these murky regions. Ground-glass opacity may indicate the presence of fluid, pus, or cells filling the air gap, thickening of the alveolar walls, or thickening of the interstitial space between the lungs [10].

Pneumonia, often known as a lung infection, may be brought on by acute respiratory illnesses. When a person who is healthy breathes in, the alveoli, which are very little sacs that make up their lungs, get filled with air. Pneumonia is characterized by the presence of pus and fluid-filled alveoli, both of which make it challenging to breathe and reduce the body's capacity to absorb oxygen [30]. Each year, there are around 200 million instances of viral community-acquired pneumonitis, with 100 million of those cases affecting kids and 100 million affecting adults [1].

To produce two-dimensional pictures of inside organs, X-rays employ radiation. Organs and tissues that absorb more radiation seem white or gray, while substances that do not (such the gases in the lungs) absorb as much radiation as they should appear black. On the chest X-ray, we can see COVID-19, lung opacity, and viral pneumonia. Despite this, due to legal constraints, chest X-ray datasets are rarely accessible. To train the algorithm and improve results and categorization, we must therefore produce false datasets.

A generative model is trained using the more reliable method created by Goodfellow et al. This method consists of two different kinds of networks: the discriminator network and the generator network. In a minimax game, two networks compete against each other while simultaneously receiving instruction.

The GAN technique is a kind of unsupervised neural network that may produce realistic graphics without depending much on original input. The problem of inadequate numbers and imbalances in medical imaging data has been addressed using picture alterations and other data augmentation procedures to prevent performance degradation. Automated medical analysis solutions may aid in lowering the expertise boundaries of the person with the relevant expertise and accelerating the process of diagnosis and treatment.

It has been shown that training classification models benefits from GAN-based data augmentation. Dermatologists must employ data augmentation techniques based on GAN to help in the classification of illnesses and help them reach more precise diagnostic conclusions. Furthermore, when using deep learning in medical diagnostics, a model's classification accuracy may be subpar, limiting the accuracy of skin lesion categorization. This is because bad data might affect a model's classification accuracy.

Some examples of simple data augmentation techniques for playing with photos include inverting, scaling, rotating, shearing, and adjusting the contrast. Improved data are created from a small sample of original data sets and have the same level of variety as the original data, hence this method of data upgrading does not significantly boost performance on its own. In order for deep neural networks to extract more nuanced qualities from a sample, they must use all of the data available in the sample.

Various GANs have been proposed to be very efficient in many applications, including computer vision. These models could give rise to fresh, logically sound synthetic samples, reducing concerns about overfitting and a lack of variety in the sample data and, thus, reducing the capacity loss brought on by skewed class proportions.

Recently, for example, some chest ailments have been diagnosed using CNNs and GANs. Symptoms and a chest X-ray can be used to figure out what this illness is. A chest X-ray can be used by radiologists to see if someone has a coronavirus infection. Patients are likely to get better, and this has led to the creation of many more deep learning models. CNNs also do well in medical imaging when they are given the right information. To get to this level of performance, millions of system parameters are adjusted and trained using labeled data. Because CNNs have so many parameters, they tend to fit small datasets too well. So, the performance of generalization is related to the amount of labeled data. Tiny datasets are the hardest thing to work with in medical imaging because they are rare and don't vary much. The GAN framework is utilized for a variety of different medical imaging modalities.

The detection of melanoma, which is one of the most dangerous kinds of skin cancer, is yet another noteworthy development in this present instance. The categorization of skin lesions using deep learning diagnostic algorithms is a challenging problem since there are not enough skin lesion photographs that have been annotated and the datasets within each class are not evenly distributed. Dermatologists need data augmentation methods based on GANs to help with skin lesion classification and give them the ability to make more accurate diagnostic decisions. Furthermore, when using deep learning in medical diagnostics, a model's classification accuracy may be subpar, limiting the accuracy of skin lesion categorization. This is because bad data might affect a model's classification accuracy. For the purpose of identifying and categorizing lesions in images, recent research has developed a novel skin framework. The foundation of this system is laid by style-based generative adversarial networks (GANs) and DenseNet201, and it is referred to as a skin lesion augmentation style-based generative adversarial network (SLA-StyleGAN). The aim of this structure is to solve the aforementioned problems.

A theoretical concept called the OT model enables users to perform quantity conservation in computer physics experiments, interpolate between devices, measure gaps between functions (or distances between bigger sized devices), and measure gaps between functions. In fact, OT is a booming therapeutic field, with active research groups concentrating on theoretical and applied topics including geometry transmission and ML. This article aims to clarify the basic ideas at the core of the most gratifying delivery concept, present the specifically concerned ideas, and most importantly, show how they connect, so that the reader will have a better knowledge of the trendy idea that supports them. Then, in order to impart a non-stop characteristic to a discrete sum of Dirac masses, we may use a semi-discrete arrangement. Studying this particular location results in a green computational method that makes use of conventional concepts from computational geometry, such Laguerre diagrams, which are an extension of Voronoi diagrams.

There have been several OT theory models put out, and they have been quite successful in a wide range of applications. Sliced-Wasserstein Flows (SWF), for instance, is based on the implicit generative modeling (IGM) formula as a function optimization problem in Wasserstein space, with the objective of identifying a measure of the probability that is as close to the distribution of the data as is feasible while maintaining a predetermined level of expressiveness. We can transform the IGM issue into an SDE simulation problem because to the SWF's unique location at the intersection of OT, gradient streams, and SDE. They have created clear connections between the algorithm's General parameters and defects as well as a time constraint for the infinite particle mode. Numerous tests have been conducted, and the results support our hypothesis.

1.2 Research Problem

Two primary parts make up the conventional GAN: the generator network and the discriminator network. The generator network creates synthetic samples, and it is the discriminator network's task to tell them apart. The two sample types are compared and contrasted to get this conclusion. Samples are generated by the generator network, which uses prior knowledge about the distribution of the target domain to make intelligent estimates. The generator network may use misleading information to create fake samples, which the discriminator network then tries to identify and reject.

In the process of optimizing the GAN, the generator makes samples that look like they could fool the discriminator, while the discriminator gets better at telling the difference between samples. So, in order to train the GAN well, we need to optimize not just one network but two.

Because the GAN has to train the networks of both the generator and the discriminator, the optimization process may be challenging and may experience a number of downsides. First, the process of training a GAN is very prone to instability and may be heavily dependent on the rivalry that takes place between the generator and the discriminator within the context of the minimax game. To put it another way, achieving a healthy balance between the generator and the discriminator is critical for the production of high-quality sample synthesis. However, during the training of the networks, the gradient descent cannot always guarantee that a satisfactory equilibrium would be reached.

For instance, during the training process, if a discriminator is provided with a strong differentiation capacity, it is possible that the generator's gradient may disappear quite soon. As a result, the optimization of the generator may not be able to get to the point where it can approach the real distribution of the target domain. According to what is indicated in [21], it is possible to train a superior generator if the discriminator is purposefully weakened. In such a scenario, it could be necessary to go through multiple iterations of trial and error, which would make the whole process of training challenging and unpredictable. Second, there is a phenomenon in GAN known as "mode collapse", which occurs when the generator has a tendency to output samples with a limited diversity of options. In order to produce identical samples, the mode collapse is caused by the generator being stuck at the same local minimum of the cost function. This is done in order to produce the same samples. This, in turn, leads to the collapse of the mode. When this occurs, the samples that are created do not have a broad enough range of characteristics to accurately reflect the whole distribution of the target domain. Since they do not adequately represent the problem, it is impossible to use them to solve the problem of the data imbalance. As a result of this, we are unable to utilize them.

As a result, the study is attempting to answer the following question:

How effective is it to use a combination of the Optimal Transport (OT) theorem and Generative Adversarial Networks (GANs) to smoothly synthesize high-dimensional medical image from low-dimensional, insufficient, and imbalanced medical imaging data while minimizing computational difficulties to observe the difference between normal, covid-19, viral pneumonia and lung opacity?

1.3 Research Goals

We created a new loss function by combining the Wasserstein Distance and Sinkhorn Divergence. We train the generator and discriminator alternately. Adjusting generator and discriminator settings uses generator and discriminator losses. The generator and discriminator must improve until the generator produces samples that the discriminator cannot distinguish from genuine data. However, such loss function has downsides. Mode Accept first. Mode collapse, in which the generator creates a restricted number of sample variants, may affect GANs. Samples are too near to the genuine thing when generator loss is high. Second, diminishing gradient. Vanishing Gradient may occur with little generator loss. This may hinder generator development. Non-convergence. GANs may be hard to train and may not converge. If the generator, discriminator, or loss function are mismatched, this may happen. Hyperparameter sensitivity ranks fourth. The learning rate and batch size may affect GAN performance. Finally, oscillation. Oscillation occurs when the generator and discriminator oscillate but never settle. If generator and discriminator capacities are too comparable, this may happen.

The key contributions of this research:

- To increase the stability and convergence of the loss function, we employed the Wasserstein distance as the loss function. The Wasserstein distance is a more consistent and reliable measure of the difference between the produced samples and the actual data than standard measurements, which is the fundamental benefit of employing it as a loss function in GANs.
- Mode collapse, when the generator only generates a few variants of the same sample, may be avoided by using Wasserstein distance as a loss function. This is so that the generator will create a broad range of samples, since the Wasserstein distance is sensitive to the number of modes in the produced samples.
- It aids in producing pictures of greater quality and accelerating convergence compared to conventional GANs.
- As an entropic regularization term, we have also employed Sinkhorn Divergence with Wasserstein Distance. It aids in stabilizing the optimization process and avoiding the overfitting issue for the loss function. The Sinkhorn Divergence is made differentiable by this regularization term, enabling the adoption of gradient-based optimization techniques like the Adam.
- It has been shown that Sinkhorn Divergence provides quicker convergence than other distance measures and is less dependent on the selection of hyperparameters.

Rest of the report is described as follows. Here, Chapter 1 has described the background information and our research goal. Chapter 2 has described literature review and related works. Chapter 3 has described the methodology and implementation process of our approach. Chapter 4 has evaluated the approach and analyzed the result and the application of Explainable AI algorithm GradCAM.

Chapter 2

Literature Review

2.1 Background Study

2.1.1 Optimal Transport Theory (OT)

OT model is a theoretical idea that allows users to interpolate between gadgets, degree gaps between functions (or distances between larger trendy gadgets), and apply quantity conservation in computer physics experiments. OT is really a thriving clinical subject, with active research groups focusing on both theoretical and practical aspects such as geometry transmission and ML. This essay tries to explain the fundamental principles at the heart of the most satisfying delivery concept, introduce the particularly concerned thoughts, and most importantly, how they link, in order for the reader to have a better understanding of the fashionable concept that underpins them. Then we may think of a semi-discrete arrangement, in which a non-stop feature is conveyed to a discrete sum of Dirac masses. Studying this specific placement yields a green computational technique that employs traditional ideas of computational geometry, such as Laguerre diagrams, which are an extension of Voronoi diagrams.

Many OT theory models have been presented, and they have had great success in a variety of applications. Sliced-Wasserstein Flows (SWF), for example, is based on the implicit generative model (IGM) formula as a function optimization problem in Wasserstein space, with the goal of finding a measure of the probability that is as close to the distribution of the data as possible while maintaining a certain level of expressiveness. The SWF is at the crossroads of OT, gradient streams, and SDE, enabling us to convert the IGM problem into an SDE simulation problem. They've given the endless particle mode a temporal restriction and established obvious relationships between the algorithm's General parameters and faults. They've carried out a number of trials and found that the findings back up our idea.

CNN is one of the most extensively used approaches for Covid-19 detection. Synthetic CXR images generated using Auxiliary Classifier Generative Adversarial Network(ACGAN) have improved the detection of Covid-19 [15].

To create a training dataset, synthetic CXR pictures are generated using a multiscale conditional generative adversarial network with a movable mask. In this study, StackGAN++ is used as the backbone structure to generate high-resolution samples. For the incremental learning of distributions on data at different sizes, StackGAN++ makes use of tree-shaped structures to contain many generators and discriminators [16].

In order to get more accurate measurements of the distances between distributions, the Wasserstein GAN [21], often known as WGAN for its abbreviated form, was created by using the Earth Mover (EM) distance. As a result of the adversarial iterative training, the distribution of the synthetic pictures will be very close to that of the genuine CT scans. The WGAN samples will be compared against real-world examples from the classes that are subsets of the texture property. An approach called WGAN-based synthetic oversampling was first developed to address the problem of data imbalance. The purpose is to fill up the gaps in statistics for underrepresented groups. The WGAN method seeks to create new samples by providing an estimate of the distributions of the minority classes in the original data. Additionally, our technique is applied to the difficult problem of tighter classification of the seven semantic aspects of the LIDC lung nodules. This is a very difficult challenge to solve.

However, the class-imbalanced issue is highly widespread in the medical area owing to the disparities in occurrence across disorders, making it difficult to design these applications. This is one of the reasons why the medical field is so competitive. The class imbalance issue that occurs on chest X-ray (CXR) data has been addressed by a number of different algorithms that are based on GAN. These algorithms have not been trained for minor ailments since small pixels cannot extract sufficient information. Because of this, these models are not prepared to handle small diseases.

2.1.2 Generative Adversarial Networks(GANs)

Goodfellow et al. developed a more reliable system for training a generative model, which is made up of two networks, one of which is a generator and the other of which is a discriminator. In a minimax game, two networks compete with each other while being taught at the same time. In order to mislead the discriminator, the generator is taught to create synthetic versions of real-world samples, while the discriminator is taught to have a high level of discriminative ability for the generator's creations [2].

The GAN method is a kind of unsupervised neural network that may be used to create realistic visuals without relying mostly on original input. To avoid performance deterioration, image modifications and other data augmentation approaches have been used to overcome the issue of medical imaging data that has inadequate numbers and an uneven distribution of those values. Automated medical analysis solutions may help reduce the human expert boundary and speed up the diagnosis and treatment processes.

GAN-based data augmentation has been shown to be advantageous in training classification models. To aid in illness categorization and enable dermatologists to make more accurate diagnostic judgments, data augmentation approaches based on GAN must be used. Furthermore, inadequate data might result in a model's classification accuracy being poor when employing deep learning in medical diagnostics, reducing the accuracy of skin lesion categorization. The primary goal of approximating the object distribution with the use of GAN is to improve the overall performance of each individual application.

2.1.3 Image Augmentation

In order to generalize well and achieve reasonable accuracy, deep learning networks need a substantial amount of training data. The fact that the image data is too small to be used in the medical field, however, means that the data is subject to regulatory limits. Image augmentation is a strategy for extending the size of the training dataset to improve the overall performance of the model and its capacity to generalize. When augmentations are employed to reduce the likelihood of deep neural networks overfitting their data, the networks are able to perform better on computer vision tasks such as object identification, segmentation, and classification.

2.1.4 Image Classification

Image classification is an important branch of computer vision that aims to detect specific types of visual structures within an input image. The goal of image classification is to determine what kinds of visual structures are present in an image. Calculating the statistical characteristics of an image's components and then organizing those characteristics into categories is the first step in the classification process. Artificial neural networks, fuzzy-sets, and expert systems are just a few examples of the modern classification methods that have been extensively used for image classification in recent years, despite the fact that they each have their own share of flaws and provide only moderate accuracy at most. In recent years, there has been a significant increase in the use of modern classification methods for image classification. It has been shown that the DCNN architecture of deep learning is an effective advanced classification technique that can effectively resolve a variety of machine learning problems. As a direct consequence of this, there is a large range of decision-theoretic approaches that may be used for the activity of image classification. Each and every classification method begins with the presumption that the image in question reflects some number of qualities, each of which may be categorized in a single and distinct way. The number of classes may be predetermined by the analyst, or else it would be possible to have the classes automatically divided into sets of prototype classes. Image categorization, in its most basic form, refers to a method used in computer vision that assigns labels to images based on what they seem to be.

2.2 Related Works

With the emergence of open-source neural network frameworks like TensorFlow and PyTorch in recent years, image augmentation and categorization have become much more manageable. TensorFlow has many implemented libraries to create dataset objects for analysis and implementation. TensorFlow is a Python-friendly open source toolkit for numerical computation that enables machine learning and the development of neural networks quicker and more accurately [25]. PyTorch is a framework for deep learning that is available as open source software. It is used for the creation and training of individualized neural networks. PyTorch was designed specifically for image categorization thanks to its many one-of-a-kind characteristics that make it an excellent choice for the task. PyTorch offers a number of pretrained models and powerful tools for image augmentation, preprocessing, and image classification [17]. In addition to this, it provides a variety of tools for visualizing and troubleshooting neural networks. PyTorch is much simpler to work with and more straightforward to troubleshoot compared to TensorFlow. Additionally, it shortens the amount of time needed for development and cuts the number of iteration cycles necessary to reach production [34]. However, TensorFlow is the superior choice for bigger projects and operations that are more sophisticated [31].

Large volumes of labeled data are required by neural networks, which are not always accessible in medical imaging. We must supplement the labeled data with altered photos and input them into our model to fix the issue. This enhancement will aid in improving picture categorization accuracy. Moreover, the transformation is applicable to train on orientation variations.

In this paper [19], they have proposed a recurrent Wasserstein Autoencoder (R-WANS) framework for generating modeling of sequential data. This framework disentangles the representation of an input sequence into static and dynamic factors—this disentangled representation data generation with style transfer. Style transfer is used in static data and unsupervised learning frameworks studies. This unsupervised representation learning is essential in machine learning. It aids in representing unstructured data into low-dimensional space from high-dimensional sequence data. It can be applied to both static (image) and dynamic (video) data. It can disentangle and identify a person from an image and perform downstream tasks such as classification, retrieval and synthetic video generation from a video with style transfer. For this reason, they have (i) minimized a Wasserstein Distance and maximize mutual information, (ii) connected those for unsupervised representation disentanglement of sequential data, (iii) incorporate a relaxed discrete latent variable to improve the disentangled learning of action on real data by proposing two sets of effective regularizes based optimal transport (OT). However, Their research has not included supervised machine learning and non-sequential data. Their research cannot build high-dimensional space from low-dimensional manifolds, which is one of our major objectives. Their algorithm adds motion on a static video or image, which can mislead to identify the actual physical problem of an affected person.

The paper's author [12] suggested a unique end-to-end non-minimax approach for training optimal transportation mapping with quadratic cost (Wasserstein-2 distance). Using their approach, the Wasserstein–2 distance or quadratic cost may be approximated. To do so, they used input CNN and a-cycle-consistency regularization. They haven't employed bias or scales for this dimension in quadratic regularizations. They have also proved theoretical bounds about the closeness of the approximation of the transport mapping and used it. This development method can be used in the transfer of colors from one picture to another, the movement of masses across latent spaces, the translation of styles from one image to another, and domain adaptation. However, They have not included any bias and scales in their developed algorithm, which could lead to the overlapping of data. Optimal transport (OT) is great for non-overlapping data sets. Additional bias and scales are helpful to overcome this overlapping problem. Their developed approach is suitable for high-dimensional data. However, it is not possible to get high-dimensional data by performing ECG, ultrasonography, and other tests on an affected or sick person. Their proposed algorithm cannot convert a low-dimensional image into a high-dimensional space, which is also a major concern.

The authors of [6] introduce WGAN, a method-based replacement for conventional GAN training. They show how this approach can be used to increase learning consistency, get rid of problems like mode collapse, and provide practical learning curves for debugging and hyperparameter searches. The relationship between learning distribution divergences and common probability intervals with respect to the Earth Mover (EM) distance is then thoroughly theorized. One of the most remarkable practical aspects of WGANs is its ability to continually estimate the EM distance by enhancing the discriminator. They have analyzed several of their trials using the loss measure, and they see this as a significant advancement above earlier efforts to train GANs with comparable capabilities. They then provide in-depth theoretical work that demonstrates the strong connections to other distribution distances and establishes the viability of the pertinent optimization problem. They demonstrate that learning low-dimensional manifold-supported distributions cannot be done at a reasonable cost using the Kullback-Leibler (KL), Jensen-Shannon (JS), and Total Variation (TV) distances. Additionally, WGAN training becomes unstable when applying large learning rates or when using a momentum-based optimizer on the critic, such as Adam (Kingma & Ba, 2014) (with 1>0).

WGAN improves ward-reliable GAN training. It may still provide inadequate samples or fail to converge, according to [8]. They found that weight clipping's Lipschitz limitation on WGAN's critic causes these complications. Next, the authors propose penalizing the critic's gradient norm for its contribution to weight reduction. This technique outperforms classic WGAN since it can train 101-layer ResNets and language models with continuous generators with less hyperparameter adjustment and a high success rate. They offer a uniform sample distribution along a straight line with a penalty coefficient and a two-sided penalty to compute the gradient penalty to meet the Lipschitz condition. They tested the model on the LSUN bedrooms dataset using six GAN structures. This strong modeling method improves massive dataset and language modeling. GANs are strong yet unstable during training. The weight limit and cost function interact, which might cause the gradient to disappear or grow overly big if the clipping threshold is not set properly, making WGAN optimization problematic. Most GAN implementations employed batch normalization for the generator and discriminator to improve training consistency. The discriminator now maps an entire set of inputs to create another set of outputs.

According to the authors of the research, early data reveal that anomalies occur in the chest X-rays of individuals with COVID-19 [15]. Consequently, various deep learning approaches have been created, as well as groundwork demonstrated that COVID-19 patient identification using chest X-rays has a high degree of accuracy. CNNs and other deep learning networks need a large quantity of training data. Because the epidemic is new, it might be difficult to quickly collect a significant number of radiographic photographs. Consequently, we created CovidGAN, an ACGAN-based model for creating synthetic chest X-ray (CXR) images for this investigation. We also illustrate how CovidGAN-generated synthetic images may be leveraged to enhance the capability of detecting COVID-19 possessed by CNN. Just using CNN resulted in an accuracy of 85 percent for the classification. Through the use of fabricated images generated by CovidGAN, the accuracy was improved to 95%. We anticipate that we will be able to discover COVID-19 more quickly and construct radiological systems with a higher degree of reliability by using this technology.

It is crucial to identify any lesions present in the medical picture data in order to provide an accurate diagnosis, choose the most effective treatment option, and assess the patient's anticipated result. Supervised learning is used by the most majority of segmentation and classification algorithms, and it relies on correctly labeled voxels or images. Contrarily, imaging may be time-consuming and technical expertise is often required for reliable lesion diagnosis. [14] The author proposes a medical image production strategy using an abnormal-to-normal translation generative adversarial network (ANT-GAN). A medical image synthesis tool, this model can take an aberrant medical image and generate a normal-looking medical image from it without the need of a training dataset. What it does is transform a medical image that once seemed aberrant into one that seems normal. Unlike traditional GANs, which aim to construct realistic samples with variations, our more constrained model tries to generate a normal-looking picture that resembles one with lesions. In contrast, traditional GANs rely on random sampling to construct realistic examples. Findings from our research suggest that providing a normal equivalent to a medical image might be helpful for medical imaging tasks such as lesion segmentation and classification. On the other hand, the ANT-GAN model may generate a lesioned picture that is otherwise indistinguishable from the original healthy image. Our research demonstrates that this is suggestive of data augmentation. An adversarial generative network called ANT-GAN was proposed to remove lesions from a medical image while maintaining its overall appearance. They also showed how having both images may help with medical image segmentation and classification by providing more information about how the image should appear if the patient were healthy. To summarize, this was crucial for maximizing the potential of this little data set in compared to other GAN configurations.

Skin cancer has become one of the world's most serious concerns due to its high incidence rate and poor survival rate. Over the last fifty years, the number of people diagnosed with skin cancer has risen rapidly. Early survival rates for melanoma patients may reach 95% within five years, but late survival rates are barely 15%. This huge disparity emphasizes the necessity of early melanoma detection and diagnosis. As a result, detecting this kind of skin cancer is critical. Skin cancer is diagnosed via dermoscopy, an imaging procedure. The naked eye's identification has a considerable degree of inaccuracy, notwithstanding its high accuracy rate. ANN is made up of artificial neurons that are modeled after real neurons in the brain. Researchers found that convolutional neural networks (CNNs), a kind of network often employed for the task of image classification, are a refined version of the more traditional feedforward neural networks [24]. The tremendous increase in computer performance has allowed for substantial progress in artificial intelligence and deep learning technologies. The field of image processing is one area where these technologies have been put to good use. Discrete convolutional neural networks (DCNNs) can reliably recognize, segment, and identify areas and objects in medical pictures. The application of deep learning is on the rise in clinical settings, where it is yielding impressive results in pattern recognition, especially in the areas of image segmentation and classification. As a result, a recent study has presented a strategy based on style-based GANs and DenseNet201 for enhancing skin blemishes. Their ability to lose has also been strengthened. This study will aid in the diagnosis of skin diseases by dermatologists, increase patients' knowledge of skin health, and provide a valuable resource for artificial intelligence-based dermoscopy picture processing. In order to better serve patients and dermatologists, they are also working to make this computer-based diagnostic process mobile-friendly.

Data scarcity and imbalance plague the healthcare industry. Due to privacy issues, most datasets are unavailable. Furthermore, there are no data-sharing incentives. Because of these difficulties, generative modeling and data augmentation have become important in that field. This work investigates a machine learning-based strategy for distinguishing between Covid-19 and non-Covid-19 impacted Chest X-Rays in this setting. Furthermore, this study looked at an unique use of variational autoencoders (VAEs). A VAE model is trained to create scan pathways, which are image-based representations of eye-tracking data. The author of the article [9] looked at the benefits of using VAEs in the case of unbalanced datasets. They did this by extracting an unbalanced subset of the well-known MNIST dataset. Furthermore, the authors added synthetic samples to a VAE model. The addition of VAE-generated samples improved classification accuracy in general, according to their findings.

Similarly, the authors of the research [13] examined the influence of various augmentation procedures on model accuracy in supervised machine learning situations. The authors concentrated on datasets that were comparably smaller. There were fewer than 1000 samples per class. Their research is based on 19 different benchmark datasets. The accuracy of their predictions has been enhanced by around 3% thanks to the use of VAE and GAN models.

The study [22] proposes an OTDDA architecture to switch between diagnostics for a machine's fault in different modes of operation. There are two major concerns that need to be taken into account with this system. First, there are difficulties in precisely gauging and identifying the gap between the origin and destination spaces. Secondly, focus on the distinctions between features as opposed to labels. Their focus is on unsupervised domain adaptation, which involves matching the distribution of the source domain to that of the target domain. This is done by first applying the classifier trained on data from the source domain to the target domain. Furthermore, the OTDDA design has been proposed to improve the accuracy of fault diagnostics during real industrial production.

Chapter 3

Methodology

Proposed OT GAN Approach for Chest X-Ray Analysis of Lung Diseases

In this chapter, the approach is illustrated in the following sequence: gathering data in order to train the Neural Network, picture pre-processing and training the Neural Network, and assessment procedures.

We have approached the following steps:

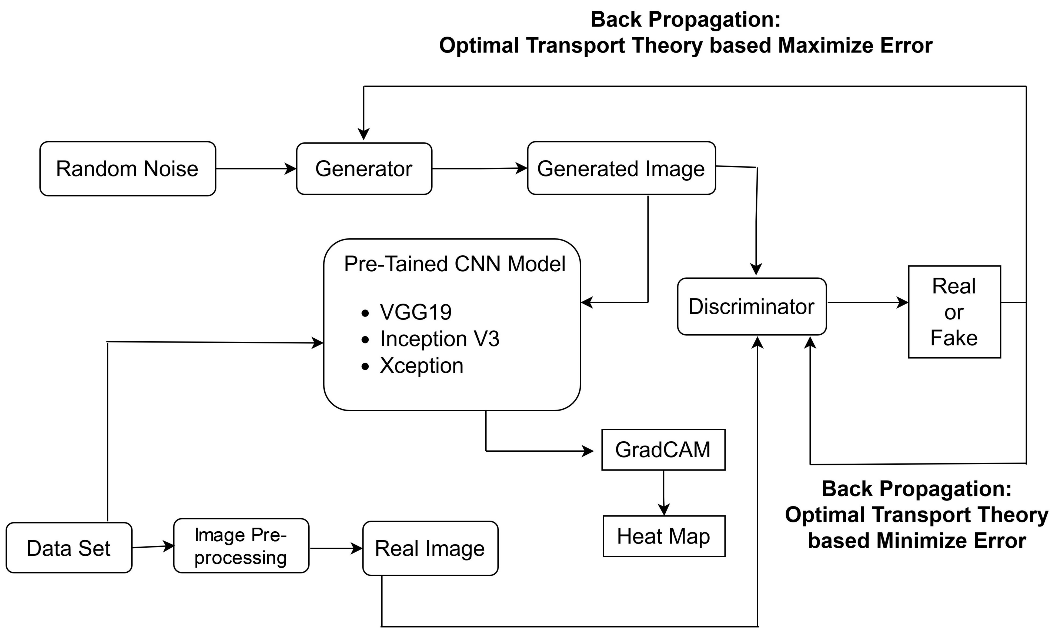


Figure 3.1: The top level overview of the proposed method

The Figure 3.1 describes the top level overview of the proposed method of the Optimal Transport based Generative Adversarial Networks for Medical Image Augmentation and Classification.

Our primary focus is on dividing the imaging data into four distinct categories: healthy, COVID-19, viral pneumonia, and lung opacity. We provide an approach to medical picture enhancement and classification using Generative Adversarial Networks (GANs) and Optimal Transport (OT) theory. It consists of many stages: 1) The generator creates a false picture out of noise in an attempt to trick the discriminator. The generator's primary purpose is to create an image that looks as realistic as possible, (2) feed a real image from our dataset into the discriminator function from our dataset, along with a fake image generated by the generator, and have the discriminator determine which image is which, (3) extract latent space from the discriminator's output. Optimal Transport (OT) theory-based maximizing error for generator and Optimal Transport (OT) theory-based minimizing of error for discriminator are used to calculate the loss function for both the genuine and false images, respectively. (4) training the generated images and real images from the dataset with three transfer learning models (VGG19, Inception V3, and Xception), each with 500 epochs, (5) analyzing the last layer of each model with Explainable AI Algorithm GradCAM to generate the heat map images and the superimposed images for evaluating the models.

The next few sections explain different phases of the proposed model in detail.

3.1 Dataset

An XR image dataset including the PA and AP perspectives (Posterior-Anterior (PA) and Anterior-Posterior (AP) views) was used to generate the research. These data sets have been acquired from publicly accessible kaggle's databases.Our Optimal Transport Theory based classification research has been conducted on two datasets, namely Chest X-ray (Covid-19 & Pneumonia)[32] and Lung Opacity Overview [11].

3.1.1 Chest X-ray Data Set

The Chest X-ray Data Set [33] has only three discriminative types of sample impressions. It comprises 6432 samples out of which 1583 are normal, 576 are COVID-19, 4273 are Viral Pneumonia. It does not carry any sample for Lung Opacity.

3.1.2 Lung Opacity Overview Data Set

The Lung Opacity Overview Data Set [11] does not carry any sample for COVID-19 and Viral Pneumonia. Rather it has 8851 ordinary samples, 9555 Lung Opacity and 11821 non-detected samples.

3.2 Image Pre-processing

The X-ray pictures were scaled in both the PA and AP directions before being sent into the neural networks. Multiclass classification use the remaining three models to determine whether the image is normal, Covid-19, has lung opacity, or is affected by viral pneumonia. For the purpose of comparing the VGG-19 and Inception V3 models, image sizes were standardized and reduced to a minimum of 224 by 224 pixels. For the Xception model, the minimum recommended picture size is 299 by

299 pixels to comply with the standard specification.

Additionally, Z-normalization was used to modify the dataset. Standardization speeds up the training process while aiding model stabilization in machine learning techniques. For z normalization, the following is appropriate (3.1).

$$\hat{X} = \frac{X[:i] - \mu i}{\sigma i} \tag{3.1}$$

Here, σi is the feature's standard deviation, and μi is the feature's mean value.

3.3 Training / Testing Data Formulation

After settling on a set of classes, we must next address the problem of data formulation. Here, we used a method called stratified random sampling. To guarantee a fair split between our training and testing data, we're using this strategy. There are now two distinct collections of data; (1) train data; (2) test data. To achieve this, you will need to choose around 80% of the data at random without replacing them and add them to your training set. You'll use the remaining 20% in your test group.

3.4 Image Classification with Transfer Learning Models

Three distinct Convolution Neural Networks were used to train, verify, and test the gathered chest X-ray data for both the binary class and multiclass setup. On two local machine versions with a Ryzen 7 3700X CPU, RTX 2060 Super 8GB GPU, and 64 GB RAM specification and an 11 GEN core i-5, RTX 3060 6GB GPU, and 24 GB RAM, the VGG19, Inception V3, and Xception models were utilized to train the dataset. Using a weighted categorical loss function helps fix the problem of class imbalance in the training data. When creating models, the Adam optimizer is used with its default parameters to ensure computational efficiency and a fast learning rate. If you set your patience to 15, for example, early stopping will be utilized to keep an eye on validity loss and end training once it has stabilized. Model training was fine-tuned using a factor of 0.25 and 15 iterations of patience until validation loss stabilized. If the model doesn't improve after 15 iterations, the training process is terminated. Mini-batch size of 32 images and 500 epochs were used to train the models, and the resultant receiver operating characteristic (ROC) curve, confusion matrix, and evaluation matrices were generated. The GradCam Method is also used to construct multi-tiered heatmaps. By superimposing the original image on top of the heatmap, we can better pinpoint the crucial regions of the picture, as well as make predictions about the pathological condition and the model's interpretability. As can be seen in Figure 3.1, the full system diagram used in this investigation is being used. Multi-class classification issues find Normal, COVID-19, Lung Opacity and Viral Pneumonia.

3.4.1 VGG19

In the 2014 ImageNet Large Scale Visual Recognition Challenge, the neural network VGGNet [3] fared very well (ILSVRC). First in image localization and second in image classification. VGG19, which comprises 16 convolutional layers, 5 Max-pooling layers, 1 average pooling, flattening, and a thick layer, performed well in our classification exercise (see Figure 3.2). The convolutional layers of a VGG19 CNN employ small 3x3 filters with a stride of 1, while the MaxPooling and Downsampling layers use 2x2 filters with a stride of 2 to form six huge blocks. We froze the weight up to the max-pooling layer in the conv5 block, added one average pooling layer, one flattening layer, and one dense layer to our specialized Vgg19 network to fine-tune the model. Figure 3.1 depicts VGG19 from our lung illness detection classification trials. The four class challenges discovered were "Normal," "Covid-19," "Lung Opacity," and "Viral Pneumonia," whereas the other two classes only detected "Covid-19" and "Normal" images.

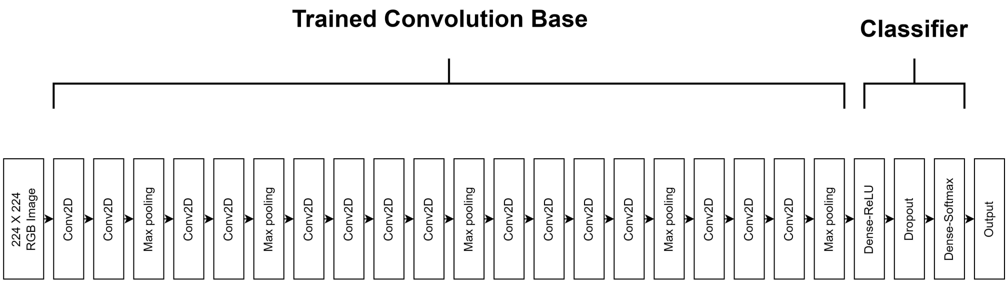
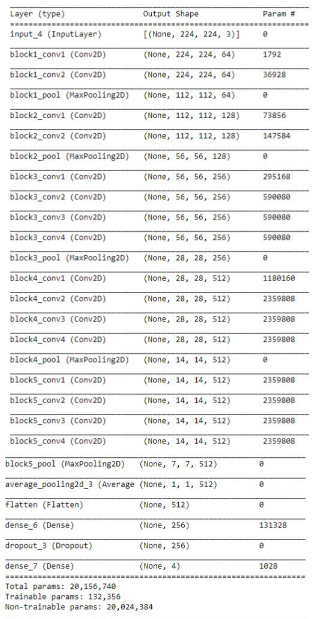


Figure 3.2: Architecture of VGG19



Layer (type)	Output Shape	Param #
input_4 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv4 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv4 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv4 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
average_pooling2d_3 (Average	(None, 1, 1, 512)	0
flatten (Flatten)	(None, 512)	0
dense_6 (Dense)	(None, 256)	131328
dropout_3 (Dropout)	(None, 256)	0
dense_7 (Dense)	(None, 4)	1028

Total params: 20,156,740
Trainable params: 132,356
Non-trainable params: 20,024,384

Figure 3.3: VGG19 model summary

3.4.2 Inception V3

In our study, we were able to recognize images with the help of a neural network called Inception v3, which is a version of the Inception family of networks and is based on depth-wise separable convolutions. In a manner comparable to VGG 16, we made adjustments to the previous three layers of the primary Inception v3 [5] model. A dropout layer, a dense layer, a flattening layer, an average pooling layer, and a dense layer were all included in the last step of the classification process. From a total of 22327396, Inception v3 allows for the modification of 524612 parameters. In all, there are 21802784 variables that cannot be altered by training.

3.4.3 Xception

Researchers at Google developed a variant of Inception-Depthwise v3's Separable Convolutions-based deep neural network architecture they called Xception [7]. In the

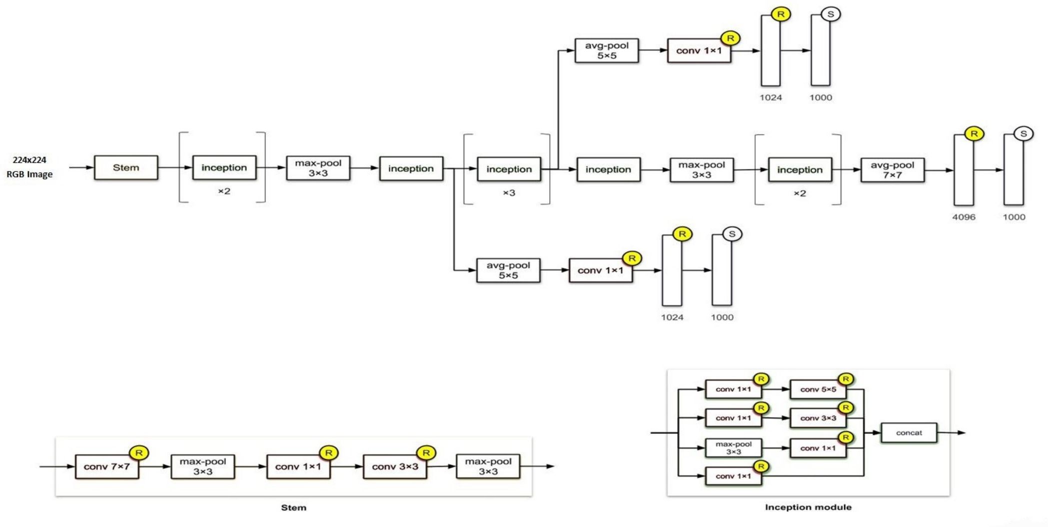


Figure 3.4: Architecture of Inception V3

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 32, 32, 32)	896
dropout_1 (Dropout)	(None, 32, 32, 32)	0
conv2d_2 (Conv2D)	(None, 32, 32, 64)	18496
dropout_2 (Dropout)	(None, 32, 32, 64)	0
max_pooling2d_1 (MaxPooling2	(None, 16, 16, 64)	0
flatten_1 (Flatten)	(None, 16384)	0
dropout_3 (Dropout)	(None, 16384)	0
dense_1 (Dense)	(None, 1024)	16778240
dropout_4 (Dropout)	(None, 1024)	0
dense_2 (Dense)	(None, 10)	10250

Total params: 16,807,882
Trainable params: 16,807,882
Non-trainable params: 0

Figure 3.5: Inception V3 model summary

convolution process known as depthwise convolution, each input channel is passed through a single convolutional filter. Pointwise convolution is a kind of convolution where the 1x1 kernel is iterated over each point. This kernel's depth is identical to the number of channels in the original picture. A pointwise convolution is of- ten used in conjunction with a depthwise convolution to build depth-wise-separable convolutions. In the modified depthwise separable convolution, the pointwise convolution still comes after the depthwise convolution, even though it came before it in the original depthwise separable convolution. Changes to the Xception model's separable convolutions were implemented in recent years. Our Xception model for labeling normal, Covid-19, Viral Pneumonia, and Lung Opacity images are shown in Figure 3.6. Separable Convolutions are modified depth wise-separable convolutions that use the intermediate flow's residual connections. As part of the Xception paradigm, information is sent from the input flow to the central flow eight times before eventually leaving the system through the exit flow. Average Layers of flattening, density, and pooling with a 4x4 pool size are used to recognize images. Images are classified as either “Normal”, “Covid-19”, “Lung Opacity”, and “Viral Pneumonia” using a final thick layer with an output size of four and similar architectural principles.

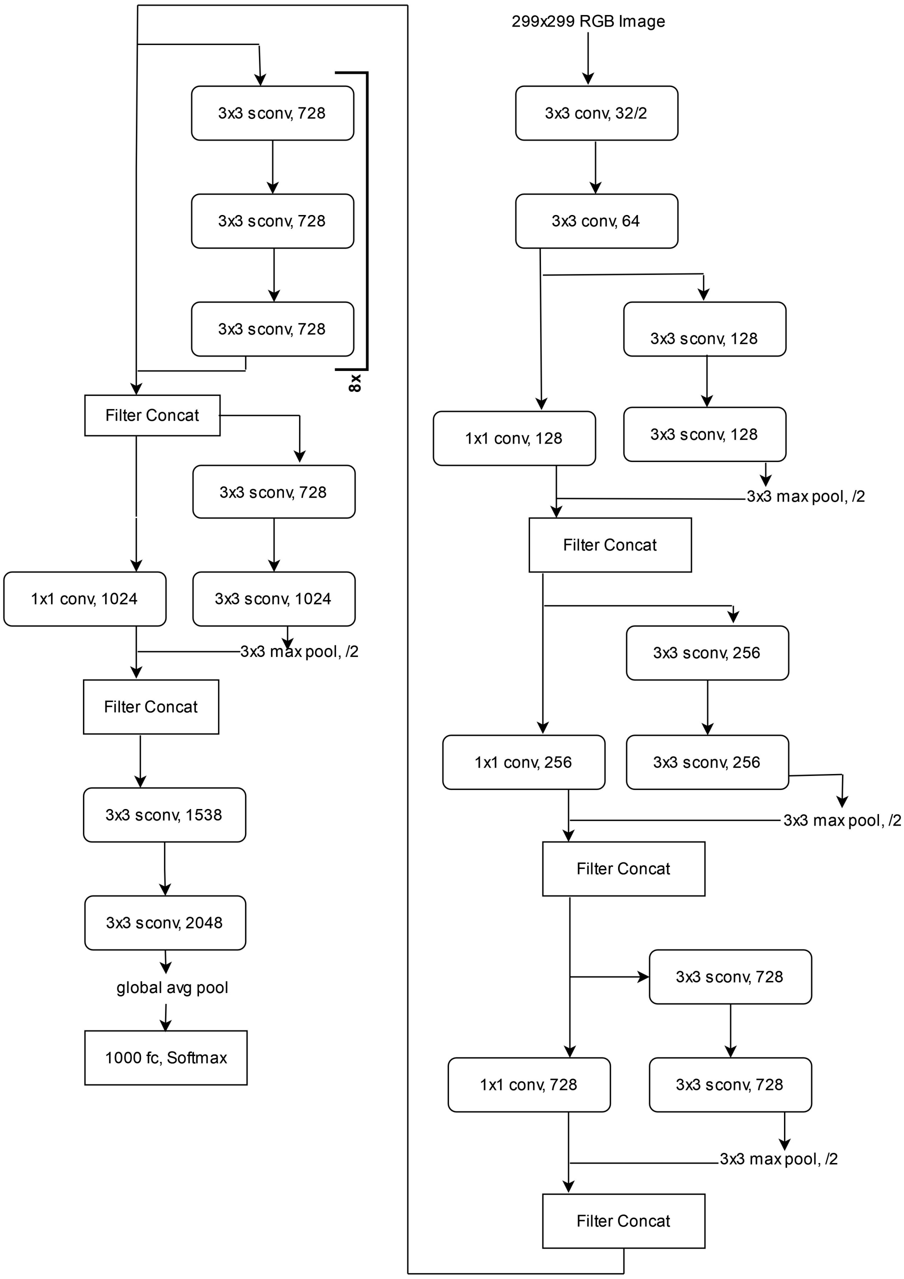


Figure 3.6: Architecture of Xception.

separable_conv2d_29 (SeparableC	(None, 19, 19, 728)	536536	re_lu_37[0][0]
batch_normalization_34 (BatchNo	(None, 19, 19, 728)	2912	separable_conv2d_29[0][0]
re_lu_38 (ReLU)	(None, 19, 19, 728)	0	batch_normalization_34[0][0]
add_10 (Add)	(None, 19, 19, 728)	0	add_9[0][0] re_lu_38[0][0]
re_lu_39 (ReLU)	(None, 19, 19, 728)	0	add_10[0][0]
separable_conv2d_30 (SeparableC	(None, 19, 19, 728)	536536	re_lu_39[0][0]
batch_normalization_35 (BatchNo	(None, 19, 19, 728)	2912	separable_conv2d_30[0][0]
re_lu_40 (ReLU)	(None, 19, 19, 728)	0	batch_normalization_35[0][0]
separable_conv2d_31 (SeparableC	(None, 19, 19, 1024)	752024	re_lu_40[0][0]
conv2d_5 (Conv2D)	(None, 10, 10, 1024)	745472	add_10[0][0]
batch_normalization_36 (BatchNo	(None, 19, 19, 1024)	4096	separable_conv2d_31[0][0]
batch_normalization_37 (BatchNo	(None, 10, 10, 1024)	4096	conv2d_5[0][0]
max_pooling2d_3 (MaxPooling2D)	(None, 10, 10, 1024)	0	batch_normalization_36[0][0]
add_11 (Add)	(None, 10, 10, 1024)	0	batch_normalization_37[0][0] max_pooling2d_3[0][0]
separable_conv2d_32 (SeparableC	(None, 10, 10, 1536)	1582080	add_11[0][0]
batch_normalization_38 (BatchNo	(None, 10, 10, 1536)	6144	separable_conv2d_32[0][0]
re_lu_41 (ReLU)	(None, 10, 10, 1536)	0	batch_normalization_38[0][0]
separable_conv2d_33 (SeparableC	(None, 10, 10, 2048)	3159552	re_lu_41[0][0]
batch_normalization_39 (BatchNo	(None, 10, 10, 2048)	8192	separable_conv2d_33[0][0]
global_average_pooling2d (Globa	(None, 2048)	0	batch_normalization_39[0][0]
dense (Dense)	(None, 1000)	2049000	global_average_pooling2d[0][0]

Total params: 22,910,480
Trainable params: 22,855,952
Non-trainable params: 54,528

Figure 3.7: Xception model summary

3.5 Decision Modelling

25 photos from each of the four classes were used to train each model (in total 100). Initially, we used these 100 photos to train the VGG19 model, the Inception V3 model, and the Xception model. We have gathered the requisite knowledge and classification data after training using these transfer learning models. Then, using the same 100 photos, we created a network of generative adversarial networks that added 1600 more images using our suggested Optimal Transport theory. The VGG19 model, the Inception V3 model, and the Xception model were all retrained using the enhanced 1600 pictures. Using these transfer learning techniques, we have once again gathered the essential knowledge and classification data. Our analysis of the data led us to suggest a more effective method for identifying lung opacity, viral pneumonia, COVID-19, and normal pneumonia.

3.5.1 Designing Generator

By integrating input from the discriminator, the generator component of a GAN learns to produce fictitious data.As a result, it improves its chances of having its output recognized as genuine by the discriminator. When training a generator, as opposed to a discriminator, the two systems must be more tightly coupled [29]. The GAN's generator training section consists of:

- Various input.
- To create a data instance from scratch, the input is fed into a network that instance from randomness.

- A network discriminator that sorts the data it collects.
- Output of a discriminator.
- Generator loss, a punishment for the generator's inability to trick the discriminator.

Upsampling is to increase the number of rows or columns (dimensions) of the image. That is why we initially take a input image of size 64×64, and gradually increase the size to 128×128, 256×256, and then to 512×512 with the help of Conv2DTranspose convolution operation, Leaky ReLU activation function. Here, we have used the kernel size of 4 and stride of 2×2 [18].

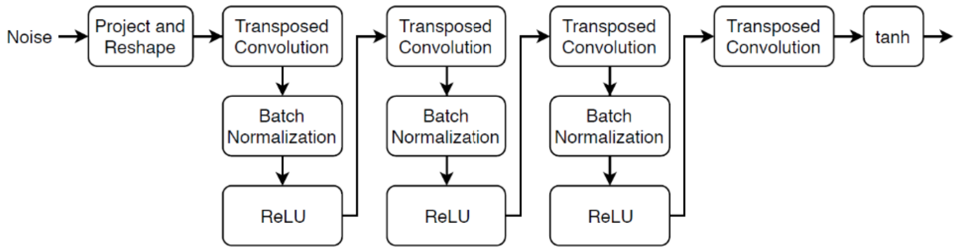


Figure 3.8: The architecture of generator neural network

3.5.2 Designing Discriminator

The training data for the discriminator originates from two sources:

- Instances of genuine data, such actual photographs of individuals. These events are used as encouraging examples by the discriminator during training.
- Examples of fake data that the generator has produced. In the course of its training, the discriminator will reference incidents like these.

Two data sources feed the discriminator While the discriminator is being trained, the generator is not being trained. Its weights remain fixed as it generates samples for the discriminator to learn on.

The discriminator is linked to two loss functions. During training, the discriminator simply uses the discriminator loss and ignores the generator loss. We use the generator loss during generator training, which is detailed in the next section [28].

When developing discriminators:

- The discriminator separates the generator's actual data from its phony data.
- The discriminator who erroneously identifies a real instance as fake or a fake instance as real incurs a discriminator loss.
- During backpropagation from the discriminator loss through the discriminator network, the weights of the discriminator change.

Here, we have taken a image of size 512×512. Then we gradually downsampled the image to 256×256, 128×128 and 64×64 with the help of conv2DTranspose, stride of size 2×2 and Leaky ReLU function.The process of downscaling reduces the size of a single photograph. The data becomes more manageable as a result. It contains:

- Decreases the data's dimensionality, allowing for speedier processing of the data (image).
- Reduces the amount of data stored.

It is sometimes mixed up with picture compression, which is entirely unrelated and has a completely different purpose. Here, we're just interested in the image's reduction in size. That effectively implies discarding part of the irrelevant data.

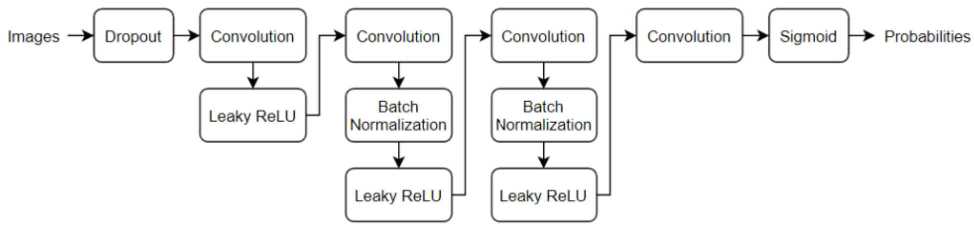


Figure 3.9: The architecture of the discriminator neural network.

3.5.3 Wasserstein Distance

When training the model, Wasserstein distance helps the generative adversarial network become more stable and offers a loss function that is correlated with the caliber of the produced pictures.

It's a big deal, and it requires a conceptual shift from the discriminator that assesses the possibility that a generated picture will be "real" to the critic model that judges the "realness" of a specific image.

We have analytically justified this conceptual shift by training the GAN using the earth mover distance, also known as the Wasserstein distance, which measures the dissimilarity between the distribution of data data in the training datasets and the distribution of data in the output instances.Mathematically it is defined as,

$$\mathbb{W}\left(\mathbb{P}_r, \mathbb{P}_g\right) = \inf_{\gamma \epsilon \Pi(\mathbb{P}_r, \mathbb{P}_g)} \mathbb{E}_{(x,y)\sim}[||x - y||] \tag{3.2}$$

Equation 3.2: Wassertein distance between distribution P_r & P_g.

In equation 3.2, $\Pi(P_r, P_g)$ is the set of all joint distributions over x and y such that the marginal distributions are equal toP_r and P_g. $\gamma(x, y)$ can be seen as the amount of mass that must be moved from x to y to transform P_r to P_g [3.2]. The Wasserstein distance is then the cost of the optimal transport plan.

$$\mathbb{W}\left(\mathbb{P}_r, \mathbb{P}_g\right) \propto \max_{\omega \in W} \mathbb{E}_{x \sim \mathbb{P}_r}[f_w(x)] - \mathbb{E}_{z \sim Z}[f_w(g\theta\left(z\right))] \tag{3.3}$$

Equation 3.3: Wasserstein Distance multiplied by constant

Instead of the discriminator that GANs have taught us to recognize, we have now added a critic. The critic network has a similar architecture to a discriminator network, but instead of maximizing the previous equation, it optimizes to find W that will minimize the Wasserstein distance. To that goal, the critic's objective role is as follows:

$$L_{critic}(w) = \max_{\omega \in W} \mathbb{E}_{x \sim \mathbb{P}_r}[f_w(x)] - \mathbb{E}_{z \sim Z}[f_w(g\theta\,(z))] \tag{3.4}$$

Equation 3.4: Critic objective function.

Here, to enforce Lipschitz continuity on the function f, the authors resort to restricting the weights w to a compact space. This is done by clamping the weights to a small range([-1e-2, 1e-2]) [27].

It is difficult to train Generative Adversarial Networks, or GANs.

A given input picture must be classified as genuine (drawn from the dataset) or false (manufactured) by the discriminator model, and new and believable images must be created by the generator model.

Since GANs need concurrent training of a generator and a discriminator model, training might seem like a zero-sum game. For steady training to occur, it is necessary to strike a balance between the capacities of the two models.

An example of a neural network trained to solve a binary classification problem, the discriminator model employs a cross-entropy loss function and a sigmoid activation function in its output layer. As a consequence, the model predicts an input's likelihood of being authentic (or fraudulent, as 1 minus the predicted) as a value between 0 and 1.

The loss function has an effect of punishing the model proportionally to the extent to which the probability distribution it predicts for a given image deviates from the predicted probability distribution. This lays the groundwork for the error that is back propagated between the discriminator and the generator, allowing for improved performance on the next batch.

Although the WGAN has a robust theoretical foundation, it differs somewhat from the conventional deep convolutional GAN in a few key ways.

The modifications are as follows:

- Utilize a linear activation function in the critic model's output layer (instead of sigmoid).
- To make the ratings for genuine and created pictures more different, train the critic and generator models using Wasserstein loss.
- After each small batch update, restrict the range of the critic model weights (e.g. [-0.01,0.01]).

Algorithm 1 WGAN, our proposed algorithm. All experiments in the paper used the default values $\alpha = 0.00005$, $c = 0.01$, $m = 64$, $n_{\text{critic}} = 5$.

Require: : α, the learning rate. c, the clipping parameter. m, the batch size. n_{critic}, the number of iterations of the critic per generator iteration.
Require: : w_0, initial critic parameters. θ_0, initial generator's parameters.

while θ has not converged **do**
  for $t = 0, ..., n_{\text{critic}}$ **do**
    Sample $\{x^{(i)}\}_{i=1}^{m} \sim \mathbb{P}_r$ a batch from the real data.
    Sample $\{z^{(i)}\}_{i=1}^{m} \sim p(z)$ a batch of prior samples.
    $g_w \leftarrow \nabla_w \left[\frac{1}{m}\sum_{i=1}^{m} f_w(x^{(i)}) - \frac{1}{m}\sum_{i=1}^{m} f_w(g_\theta(z^{(i)}))\right]$
    $w \leftarrow w + \alpha \cdot \text{RMSProp}(w, g_w)$
    $w \leftarrow \text{clip}(w, -c, c)$
  end for
  Sample $\{z^{(i)}\}_{i=1}^{m} \sim p(z)$ a batch of prior samples.
  $g_\theta \leftarrow -\nabla_\theta \frac{1}{m}\sum_{i=1}^{m} f_w(g_\theta(z^{(i)}))$
  $\theta \leftarrow \theta - \alpha \cdot \text{RMSProp}(\theta, g_\theta)$
end while

Figure 3.10: Wasserstein Distance Algorithm.

3.5.4 Wasserstein Distance based Loss Function

Because the discriminator does not actively classify examples, this loss function relies on a variant of the GAN technique. This system generates a numerical value for each occurrence. As this number might be either less than one or more than zero, we can't use it as a cutoff for determining whether or not an instance is real. Discriminator training, in its simplest form, aims to boost output for true instances while simultaneously decreasing output for erroneous ones.

Due to its inability to accurately discriminate between authentic and fake content, the discriminator is more accurately referred to as a critic. This variation is theoretically important, but in practice it may simply be seen as a recognition that loss function inputs need not necessarily be probabilities.

The loss functions themselves are deceptively simple:

Critic Loss: D(x) - D(G(z))

To optimize this function, the discriminator endeavors. It aims to increase the difference between the output on actual instances and the output on dummy instances.

Generator Loss: D(G(z))

The generator attempts to maximize this function. That is, it seeks to improve the discriminator's performance on its sham data by feeding it more of the latter.

Within these roles:

- D(x) is the critic's output for a real instance.
- G(z) is the generator's output when given noise z.

- D(G(z)) is the critic's output for a fake instance.
- The output of critic D does not have to be between 1 and 0.

The goal of the Wasserstein loss function is to widen the difference between the scores for actual and produced pictures.

We can summarize the function as it is described in the paper as follows:

- Critic Loss = [average critic score on real images] – [average critic score on fake images]
- Generator Loss = -[average critic score on fake images]

In this instance, the kind of loss is irrelevant as long as the loss for genuine photographs is low and the loss for false images is high. The Wasserstein defeat motivates the critic to disentangle these figures.

Alternately, we may encourage the reviewer to give genuine photographs high ratings and phony images low ratings in order to accomplish the same outcome. This modification is made in certain implementations.

This positive or negative split of training-acquired scores is unnecessary; just greater and smaller is sufficient, but it is a helpful way to think about it. A negative score should be given to authentic photos and a good score to fake ones.

- **Small Critic Score (e.g.¡ 0): Real – Large Critic Score (e.g.>0): Fake**

To guarantee the gradient is heading in the appropriate direction, i.e. toward diminishing loss, it is possible to multiply the average projected score by -1 when dealing with fake images. For instance, while computing weight updates, average false picture scores of [0.5, 0.8, and 1.0] over three batches of fake photos would change to [-0.5, -0.8, and -1.0].

- **Loss For Fake Images = -1×Average Critic Score**

Genuine scores do not need modification since we aim to promote lower average ratings for real photographs.

- **Loss For Real Images = Average Critic Score**

The loss function may be implemented by multiplying the anticipated label by the average score, with an expected result objective of -1 for false photos and 1 for actual images. A greater forecast average will be encouraged by the -1 label multiplied by the average score for false photos, while a smaller predicted average will be encouraged by the +1 label multiplied by the average score for actual photographs [26], [4], [23].

- **Wasserstein Loss = Label × Average Critic Score**

Or

- **Wasserstein Loss(Real Images) = 1 × Average Predicted Score**
- **Wasserstein Loss(Fake Images) = -1 × Average Predicted Score**

3.5.5 Sinkhorn Divergence

A technique for gauging the separation between probability distributions is the Sinkhorn divergence, commonly referred to as the Sinkhorn distance.Keep in mind that the regularized version of the Sinkhorn divergence is the Wasserstein distance, which is often used to measure the distance between probability distributions. The Wasserstein distance is a tool for calculating the cost of relocating a distribution from one point to another, making it applicable to optimization problems.The Wasserstein distance is noise-sensitive and challenging to calculate for large distributions. The Sinkhorn divergence resolves both problems by utilizing an approach called iterative scaling to approximate the solution and by adding a regularization factor that aids in stabilizing the optimization. The sinkhorn divergence, also known as Sinkhorn Loss or Sinkhorn Regularization, is utilized as an optimization target in many deep-learning based tasks because it approximates the optimum transit between two measures, overcomes computational challenges, and may be relaxed to a differentiable form.

Formally, the Sinkhorn divergence is described as

$$\mathcal{W}_{\varepsilon}(\mu, \upsilon) = \inf_{\pi \in \Pi(\mu,\nu)} \left[\left(\int_{x \times x} c\,(x, y)\, d\pi\,(x, y) + \epsilon\, H\,(\pi) \right) \right] \tag{3.5}$$

where ϵ >0 is a coefficient and the entropic regularization $H\,(\pi)$ is given below,

$$H\,(\pi) = \log \left(\frac{d\pi}{d\mu d\nu}\,(x, y) \right) \tag{3.6}$$

To ensure the unique minimizer, the entropic regularization renders the Sinkhorn divergence strictly convex. Using simply matrix-vector products, the Sinkhorn divergence permits the implementation of a very simple iterative method that converges fast to a solution of (3.5). It should be noted that the sample space defines the Sinkhorn divergence (3.6) [20].

Chapter 4

Performance Evaluation and Result Analysis

In this section of the thesis, we will describe the findings and conclusions of our research.

4.1 Performance Evaluation Matrics

In order to assess the performance of three deep learning algorithms for classifying X-ray images according to (4.1), (4.2), (4.3), (4.4), and (4.5), we measured accuracy, sensitivity or recall, specificity, positive predictive value (PPV), and F1 score. The proportion of correctly identified positive samples to all positive samples is known as precision[1]. A poor accuracy will produce a large number of false positives, which means patients will be incorrectly labeled as having a certain ailment. The recall[1] is equal to the division of the number of true positives and summation of true positives, and false negatives. Recall takes into account the false negative rate. Low recollection rates might result in inaccurate diagnosis. The harmonic mean of recall and accuracy is the F1- score, which is 4.5. In the classification job, a high F1 score is sought.

$$Accuracy_i = \frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i} \tag{4.1}$$

$$Precision_i = \frac{TP_i}{TP_i + FP_i} \tag{4.2}$$

$$Sensitivity_i = \frac{TP_i}{TP_i + FN_i} \tag{4.3}$$

$$Specificity_i = \frac{TN_i}{TN_i + FP_i} \tag{4.4}$$

$$F1\_score_i = 2 \times \frac{Precision_i \times Sensitivity_i}{Precision_i + Sensitivity_i} \tag{4.5}$$

Where,

- Covid and Normal for classification problem (i)
- True Positive (TP)
- False Negative (FN)
- True Negative (TN)

This article used the gradient weighted Class Activation Mapping (GradCAM) method to visualize the input areas crucial to the model's predictions and to explain the model's efficacy via a heatmap of the three layers of a convolutional neural network (CNN). The whole Gradcam analysis procedure for normal, COVID-19, Viral Pneumonia, and Lung Opacity pictures is shown in Figure 4.22, and Figure 4.23 of our study. The identical design is used for all four classes, with the exception that Covid, Normal, Lung Opacity, and Viral Pneumonia are the output classes. It is often feasible to show the reasoning behind the model's conclusion for a given class by using heatmaps. Before developing the picture's class prediction, we put a picture through the model to acquire a forecast of what the picture will look like. The gradient of the class with regard to Feature Map activation A^k was then calculated.

$$A^k = \frac{\partial y^c}{\partial A^k_{ij}} \tag{4.6}$$

These gradients flowing back are combined globally to provide the neuron significance weights across the width and height dimensions (indexed by i and j, respectively).

$$w^c_k = \frac{1}{Z} \sum_j \sum_i \frac{\partial y^c}{\partial A^k_{ij}} \tag{4.7}$$

Next, we calculate the Grad Cam using the equation x.

$$L^c_{Grad^{AM}_C} = ReLU \left(\sum_k w^c_k A^k \right) \tag{4.8}$$

By combining the original picture with the heatmap, we produced a visualization.This letter investigated the GradCam visualization for the binary class for all transfer learning models and compared heatmaps across all the models to discover improved effectiveness and explain which portion of the features impacting model to determine a certain class. In order to determine which areas impact categorization using GradCam, we first examined various layers and blocks of a model. In order to better understand model performances, we next examined the final convolution layer of activation maps for three CNN models. For the multiclass classification, we examined the heatmaps of the final CNN layers for the three models and came to a conclusion by contrasting them.

4.2 Experimental Result

Models were evaluated on a test set that they had no access to while being trained. Testing precision, recall, F1-score (F1), accuracy (Acc), Positive Predictive Value (PPV), and Area Under the ROC Curve (AUC) were calculated to statistically evaluate the models (AUC). In Table 4.1, we can get a summary of how the three CNN networks tested in this article fared in a two-class configuration. Moreover, the three developed networks' performance is demonstrated for two varying input picture sizes. We can see that the VGG19 model has an accuracy of 96.94%, whereas Inception V3 and Xception both have an accuracy of 95.35 and 92.62 percent respectively. Precision for Inception V3 is 99.83 percent, Xception is 95.7 percent, and VGG19 is 96.1 percent. The VGG19 model has a recall of 97.70%, whereas the Inception V3 model and the Xception model both have sensitivity values of 91.56%.

Table 4.1: Performance measures for the three models

Input Image Size	Models	Accuracy	Class	Precision (PPV)	Recall (Sensitivity)	F1 Score	AUC	Explainable AI
224×224	VGG19	0.9694	Normal	0.9819	0.9609	0.9614	0.99	Y
			Covid-19	0.9819	0.9741	0.9779	1.00	
			Viral Pneumonia	0.9425	0.9719	0.957	0.99	
			Lung Opacity	0.9862	0.9804	0.9833	1.00	
299×299	Xception	0.9262	Normal	0.9102	0.912	0.912	0.95	Y
			Covid-19	0.9555	0.9532	0.9532	0.96	
			Viral Pneumonia	0.8988	0.9075	0.9075	0.92	
			Lung Opacity	0.9666	0.958	0.958	0.98	
224×224	Inception V3	0.9535	Normal	0.9983	0.9552	0.9552	0.99	Y
			Covid-19	0.9781	0.9517	0.9517	0.98	
			Viral Pneumonia	0.950	0.9474	0.9474	0.98	
			Lung Opacity	0.9795	0.9785	0.9785	1.00	

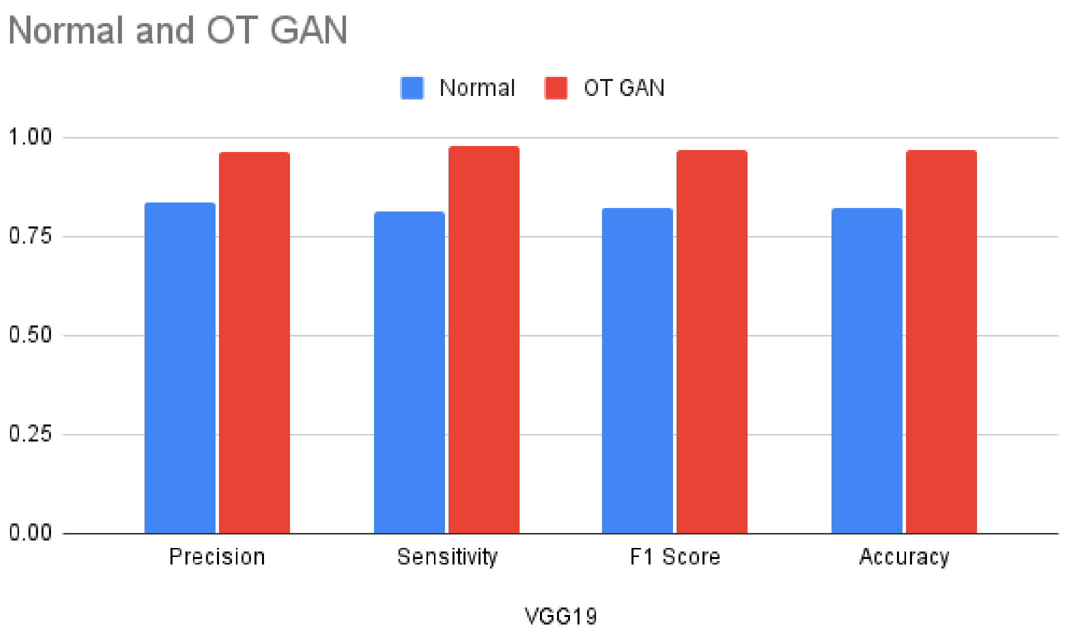


Figure 4.1: Precision, Recall, F1 Score and Accuracy bar chart comparison for the VGG19 model

In the above Figure 4.1, the bar chart compares the Precision, Sensitivity, F1 Score and Accuracy of before and after implementing the Optimal Transport theory for the

VGG19 port Theory as the previous Precision, Sensitivity, F1 Score and Accuracy was 0.8335, 0.8131, 0.8241 and 0.8202 and now it has increased to 0.9614, 0.977, 0.9691 and 0.9694 respectively.

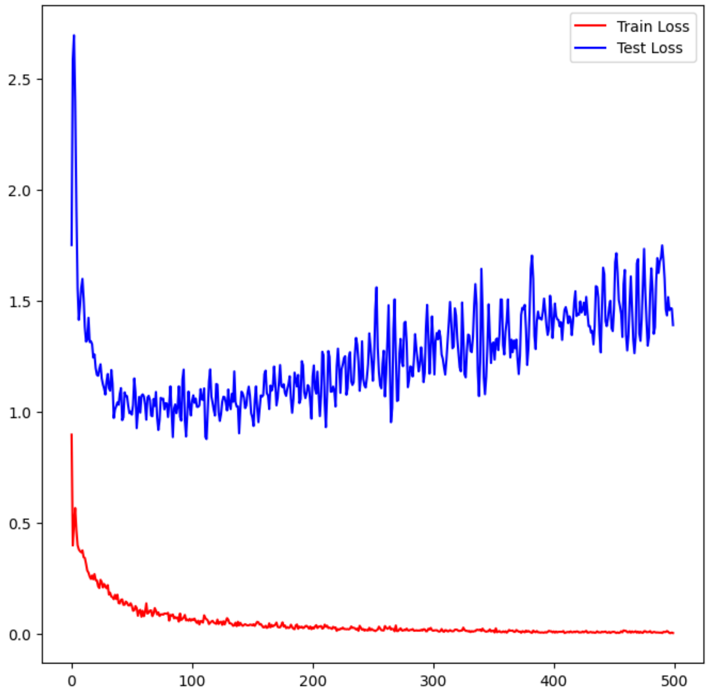


Figure 4.2: Loss Function Graph of the VGG19 Model without OT-GAN

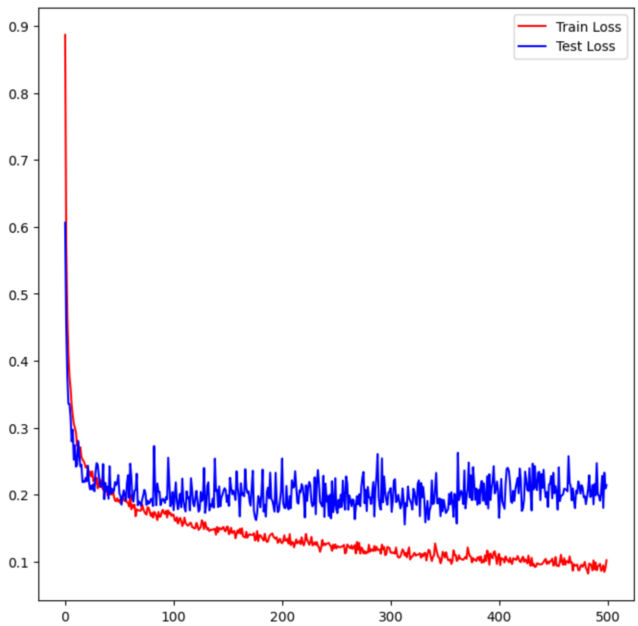


Figure 4.3: Loss Function Graph of the VGG19 Model with OT-GAN

In Figure 4.2 and Figure-4.3, we can observe that the loss value has significantly reduced with the implementation of the Optimal Transport Theorem in the VGG19 model. The graph of loss function with Optimal Transport Theorem[Figure 4.3] has significantly stabilized. Furthermore, it has also reduced the oscillation in the graph.

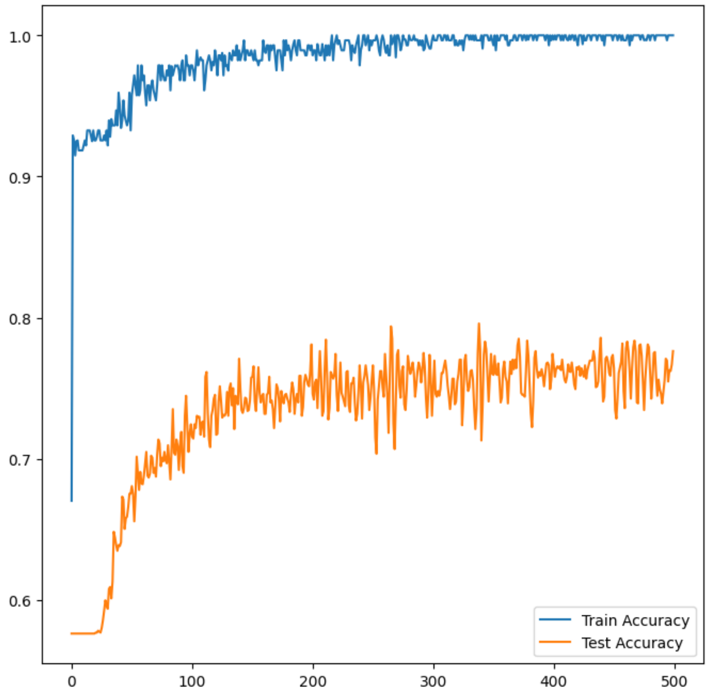


Figure 4.4: Accuracy Graph of the VGG19 Model without OT-GAN

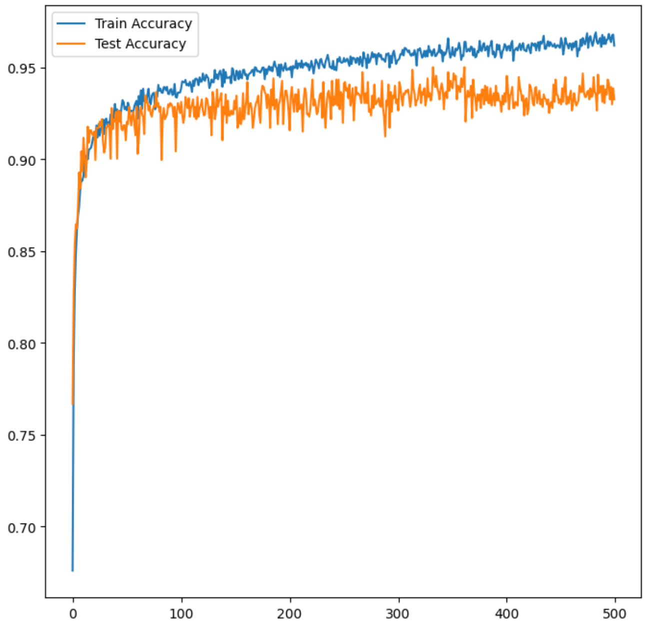


Figure 4.5: Accuracy Graph of the VGG19 Model with OT-GAN

Implementation of Optimal Transport Theory has increased the accuracy of the VGG19 model significantly. Where the previous accuracy was 82.02 percent[Figure 4.4], after the implementation of Optimal Transport Theory, it has been increased to 96.94 percent[Figure 4.5].

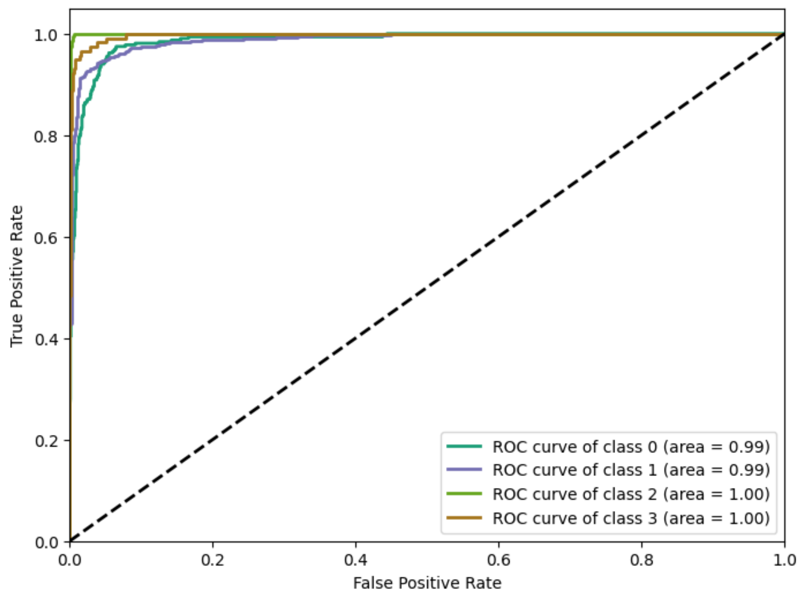


Figure 4.6: ROC Curve for the VGG19 Model

Figure 4.6 shows the ROC curve for the VGG19 model where AUC for the normal class, the COVID-19 class, the Viral Pneumonia and the Lung Opacity class are 0.99, 1.00, 0.99 and 1.00 respectively.

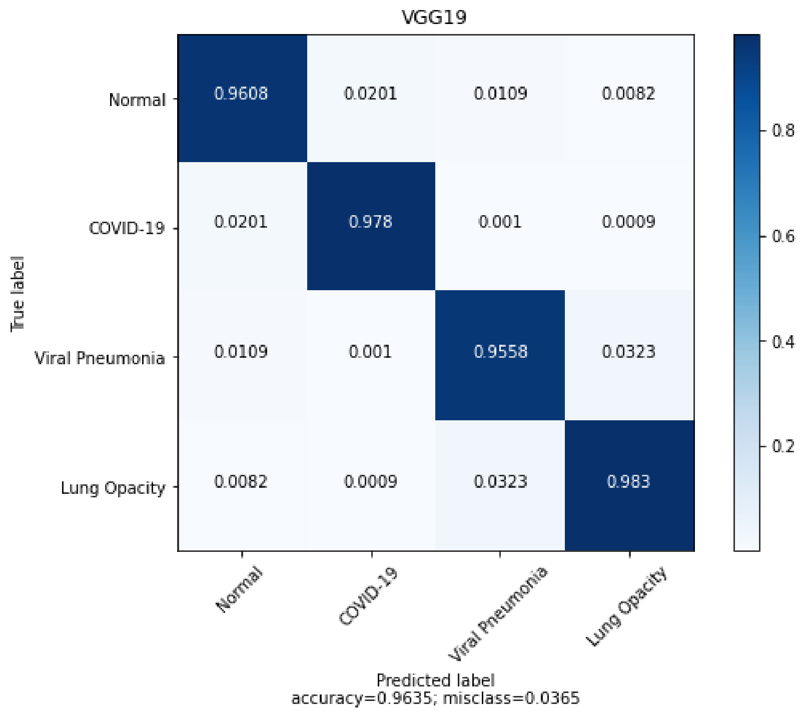


Figure 4.7: Confusion matrix of the VGG19 Model

Figure 4.7 shows the confusion matrix for the VGG19 model. It shows the performance and usefulness of our VGG19 model. From the figure[fig 4.7], it shows that the accuracy for the COVID-19 class is 0.978 where for the Normal class, the Viral Pneumonia class and the Lung Opacity class are 0.9608, 0.9558 and 0.983 respectively.

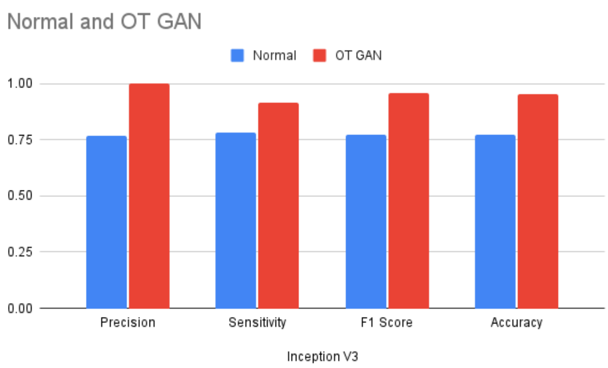


Figure 4.8: Precision, Recall, F1 Score and Accuracy bar chart comparison for the Inception V3 model

In the above Figure 4.8, the bar chart compares the Precision, Sensitivity, F1 Score and Accuracy of before and after implementing the Optimal Transport theory for the Inception V3 model. The value of each parameter has increased significantly with the implementation of Optimal Transport Theory like the previous bar chart graph as as the previous Precision, Sensitivity, F1 Score and Accuracy was 0.7655, 0.7821, 0.7737 and 0.7718 respectively and now it has increased to 0.9983, 0.9156, 0.9552 and 0.9535 respectively[fig 4.1].

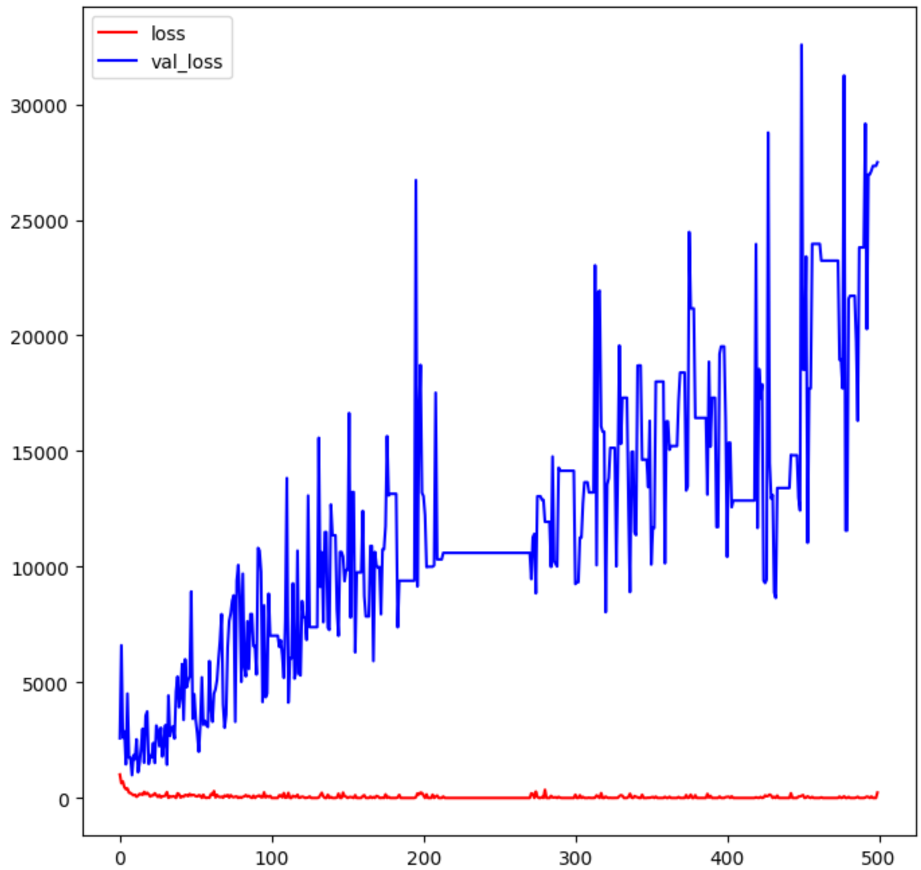


Figure 4.9: Loss Function Graph of the Inception V3 Model without OT-GAN

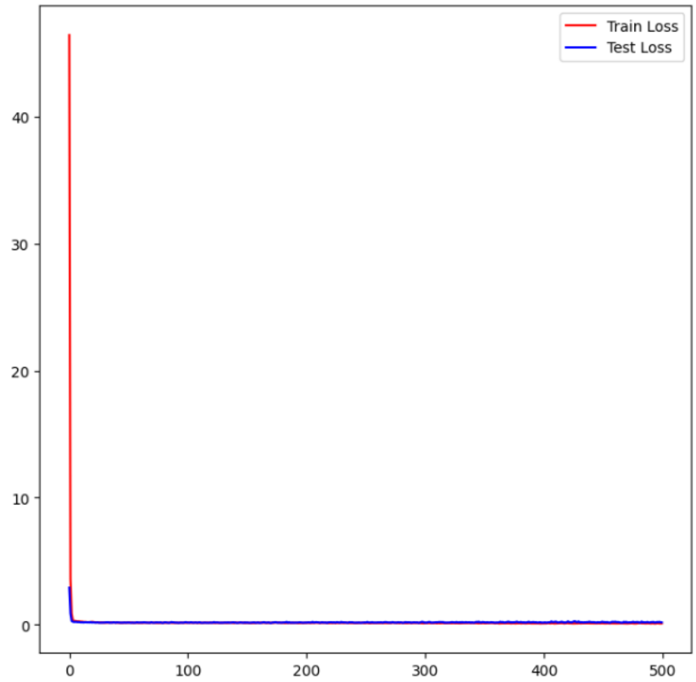


Figure 4.10: Loss Function Graph of the Inception V3 Model with OT-GAN

In Figure 4.9 and Figure 4.10, we can observe that the loss value has significantly reduced with the implementation of the Optimal Transport Theorem in the Inception V3 model. The graph of loss function with Optimal Transport Theorem[Figure 4.10] has significantly stabilized. Furthermore, it has also reduced the oscillation as well as before.

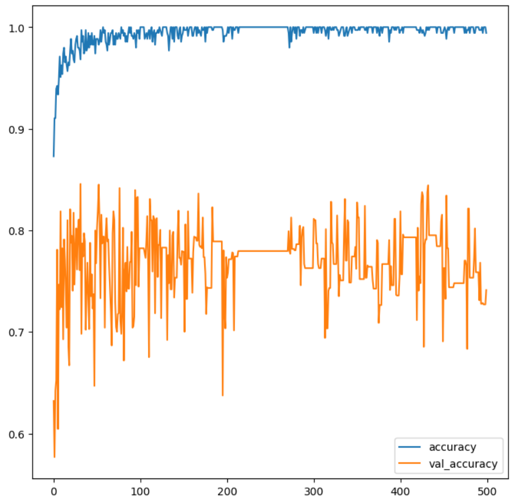


Figure 4.11: Accuracy Graph of the Inception V3 Model without OT-GAN

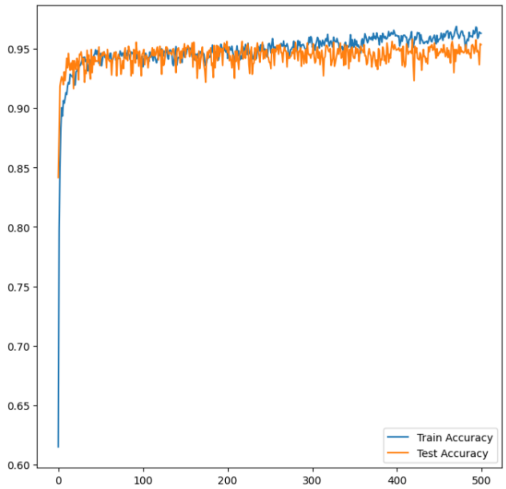


Figure 4.12: Accuracy Graph of the Inception V3 Model with OT-GAN

Implementation of Optimal Transport Theory has increased the accuracy of the Inception V3 model significantly. Where the previous accuracy was 77.18 percent [Figure 4.11], after the implementation of Optimal Transport Theory, it has been increased to 95.35 percent[Figure 4.12].

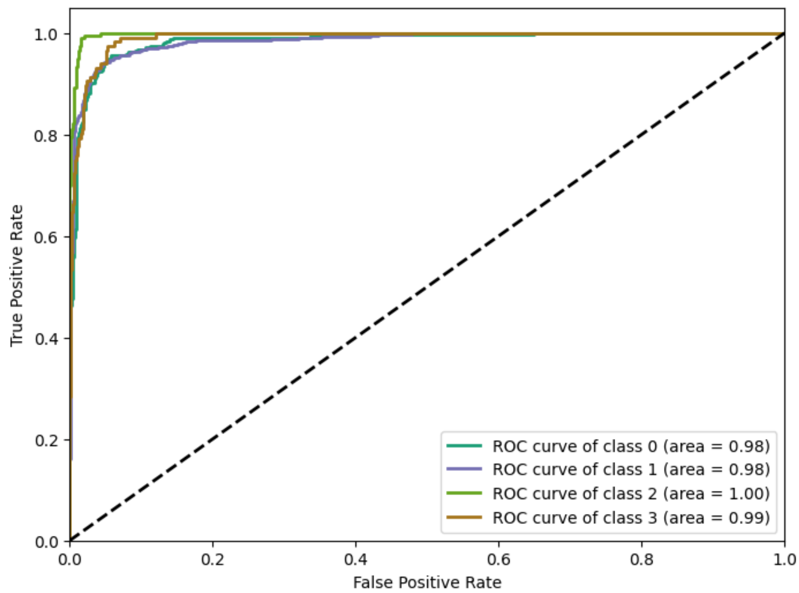


Figure 4.13: ROC Curve for the Inception V3 Model

Figure 4.13 shows the ROC curve for the Inception V3 model where AUC for the normal class, the COVID-19 class, the Viral Pneumonia and the Lung Opacity class are 0.98, 0.98, 1.00 and 0.99 respectively.

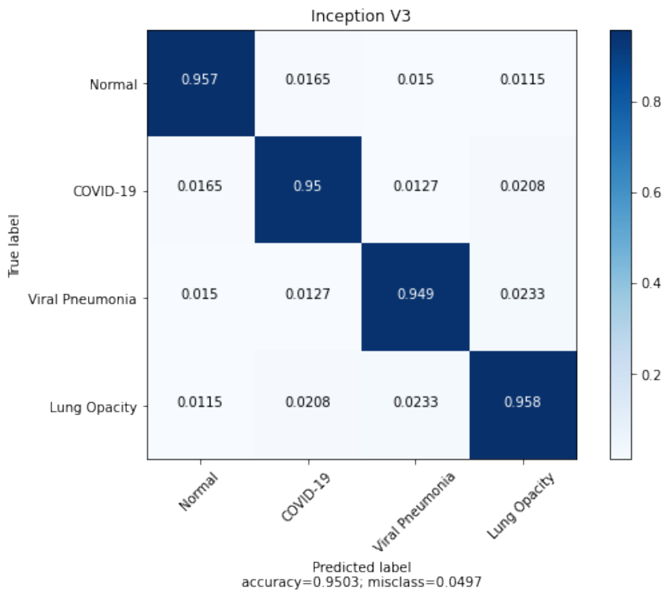


Figure 4.14: Confusion Matrix for the Inception V3 Model

Figure 4.14 shows the confusion matrix for the Inception V3 model. It shows the performance and usefulness of our Inception V3 model. From the figure[fig 4.14], it shows that the accuracy for the COVID-19 class is 0.95 where for the Normal class, the Viral Pneumonia class and the Lung Opacity class are 0.957, 0.949 and 0.958respectively.

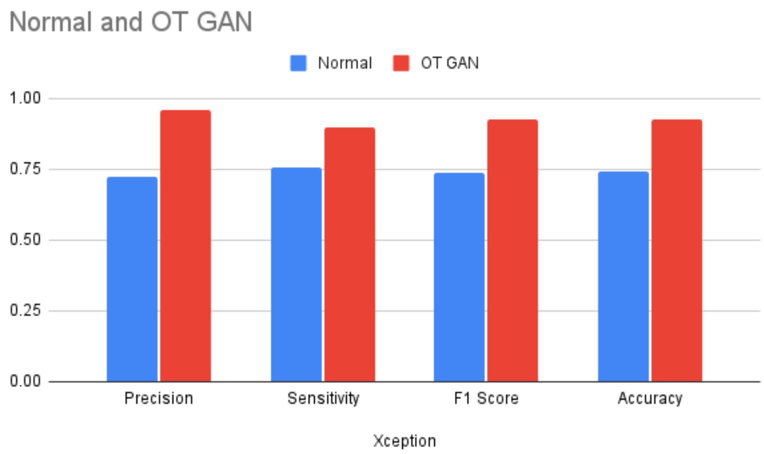


Figure 4.15: Precision, Recall, F1 Score and Accuracy bar chart comparison for the Xception model

In the above figure [Figure 4.15], the bar chart compares the Precision, Sensitivity, F1 Score and Accuracy of before and after implementing the Optimal Transport theory for the Xception model. The value of each parameter has increased significantly with the implementation of Optimal Transport Theory like the previous bar chart graphs[Figure 4.1], [Figure 4.8] as the previous Precision, Sensitivity, F1 Score and Accuracy was 0.7232, 0.7545, 0.7385 and 0.74 respectively and now it has increased to 0.9571, 0.8993, 0.9273 and 0.9262 respectively.

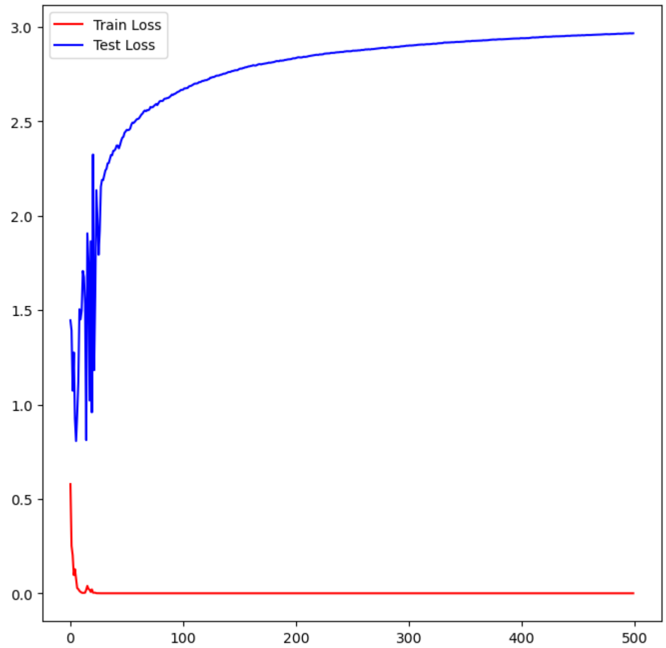


Figure 4.16: Loss Function Graph of the Xception Model without OT-GAN

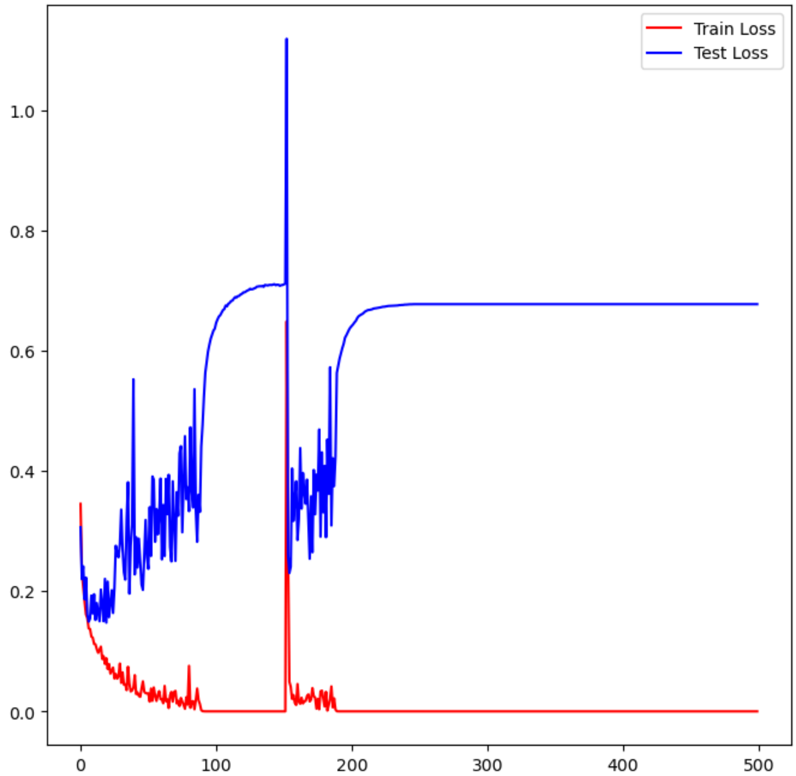


Figure 4.17: Loss Function Graph of the Xception Model with OT-GAN

In Figure 4.16 and Figure 4.17, we can observe that the loss value has significantly reduced with the implementation of the Optimal Transport Theorem in the Xception model. The graph of loss function with Optimal Transport Theorem[Figure 4.17] has significantly stabilized. Furthermore, it has also reduced the oscillation as well as before.

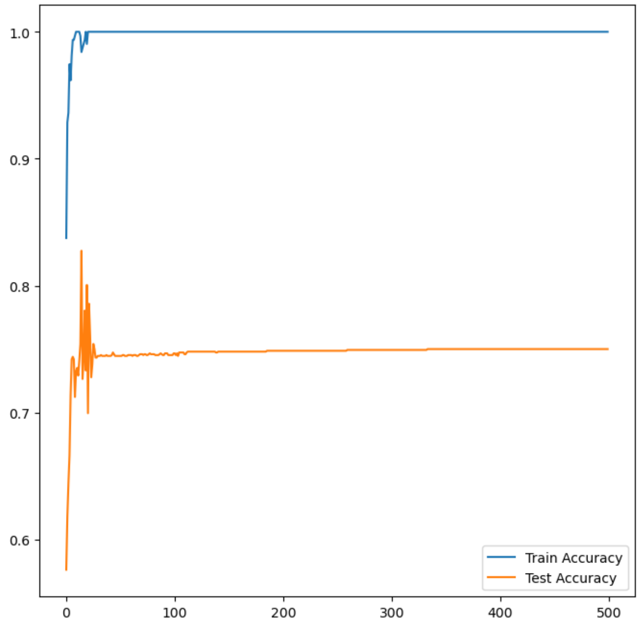


Figure 4.18: Accuracy Graph of the Xception Model without OT-GAN

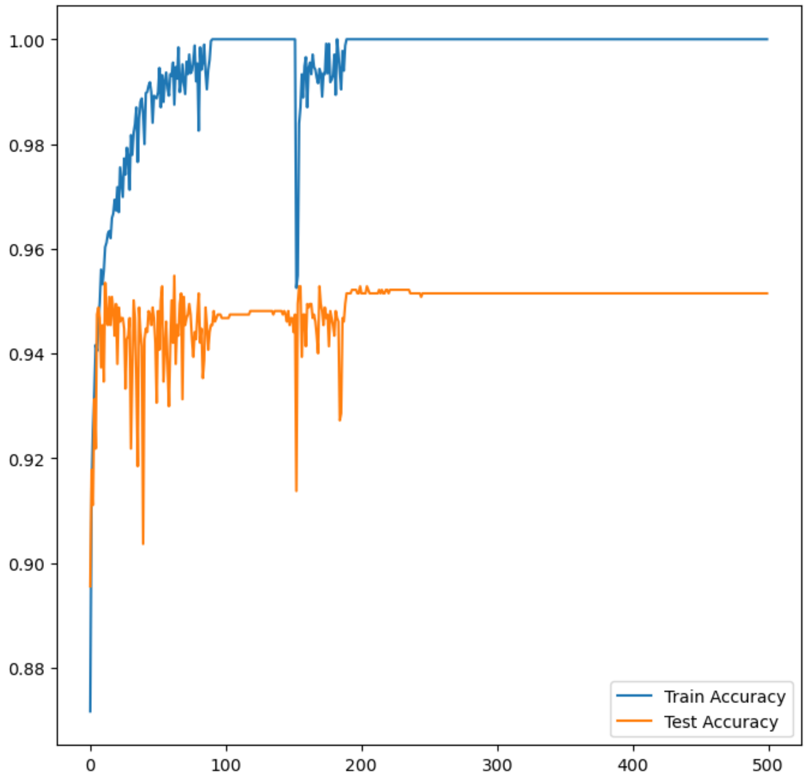


Figure 4.19: Accuracy Graph of the Xception Model with OT-GAN

Implementation of Optimal Transport Theory has increased the accuracy of the Xception model significantly. Where the previous accuracy was 74.00 percent [Figure 4.18], after the implementation of Optimal Transport Theory, it has been increased to 0.9262 percent[Figure 4.19]

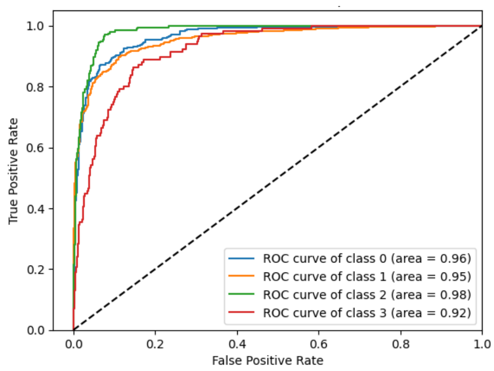


Figure 4.20: ROC Curve for the Xception Model

Figure 4.20 shows the ROC curve for the Xception model where AUC for the normal class, the COVID-19 class, the Viral Pneumonia and the Lung Opacity class are 0.96, 0.95, 0.98 and 0.92 respectively.

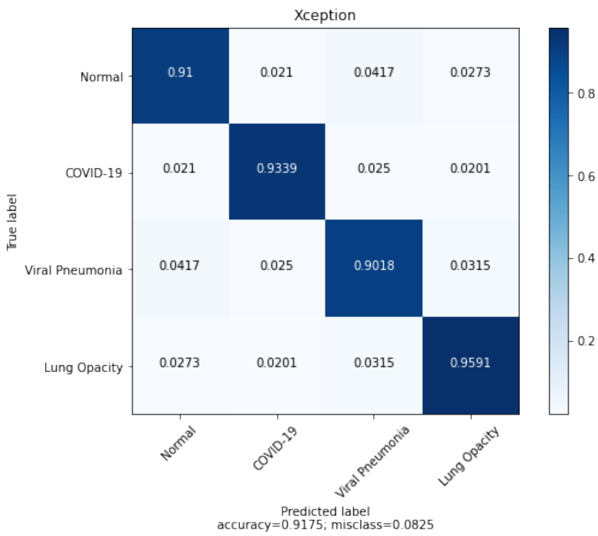


Figure 4.21: Confusion Matrix for the Xception Model

Figure 4.21 shows the confusion matrix for the Xception model. It shows the performance and usefulness of our Inception V3 model. From the figure[fig 4.21], it shows that the accuracy for the COVID-19 class is 0.9339 where for the Normal class, the Viral Pneumonia class and the Lung Opacity class are 0.91, 0.9018 and 0.9591 respectively.

4.3 Model Explainability through XAI

It is often possible to use a heatmap to demonstrate how the model came to decide a certain class. Using GradCAM activation maps and heat maps shown at various levels of three suggested networks, we qualitatively assessed network-identified areas of interest.The original picture and heatmap are stacked to highlight the image's critical parts in order to forecast the pathological condition, model interpretability, and to do further analysis. Since the final CNN layer contains high-level information, it is often used in the explainable AI algorithm,GradCAM, but we exhibited all the network layers to probe the learning process.

The GradCAM technique has been used in this work to

- To determine a model's decision-making processes, display and contrast the model's various layers.
- Pinpoint the network segments that have the most influence on classification.
- To compare the final convolutional layer of three CNN models' activation maps in order to clarify all model performances.

To further investigate which model might fare best under clinical situations, we applied explainable AI to the last layer of VGG19, Inception V3, and Xception and compared their activation maps. To compare the final convolution layer output of the VGG19, Inception V3, and Xception models, we used GradCam analysis for four class categorization.

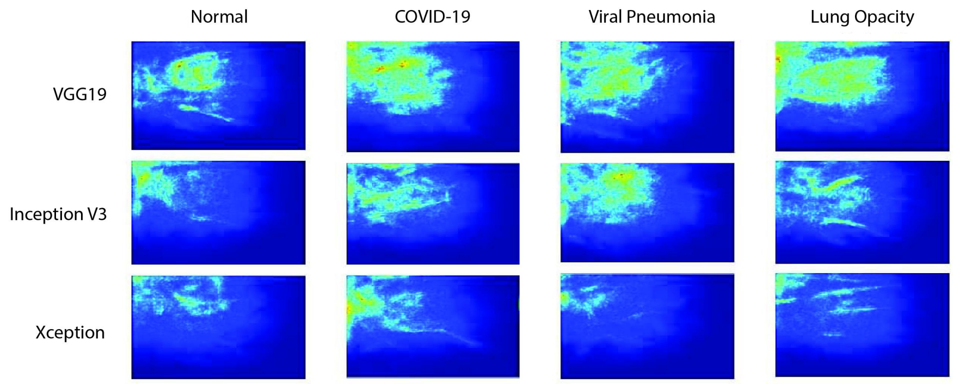


Figure 4.22: Heat Map images form GradCAM Analysis at the last layer of the CNN models

In Figure 4.22, first row: four heat map images of the last layer of the normal class, the COVID-19 class, the Viral Pneumonia class and the Lung Opacity class from the VGG19 model. The second row and the third row contain the heat map of the normal class, the COVID-19 class, the Viral Pneumonia class and the Lung Opacity class from the Inception V3 model and the Xception model respectively. In the same way, the first column contains the heat map images of the normal class generated in the last layer of the VGG19 model, the Inception V3 model and the Xception model. The second column contains the heat map images of the COVID-19 class generated in the last layer of the VGG19 model, the Inception V3 model and the Xception model. The third column and the fourth column contain the Viral Pneumonia class and the Lung Opacity class generated by the same transfer models respectively.

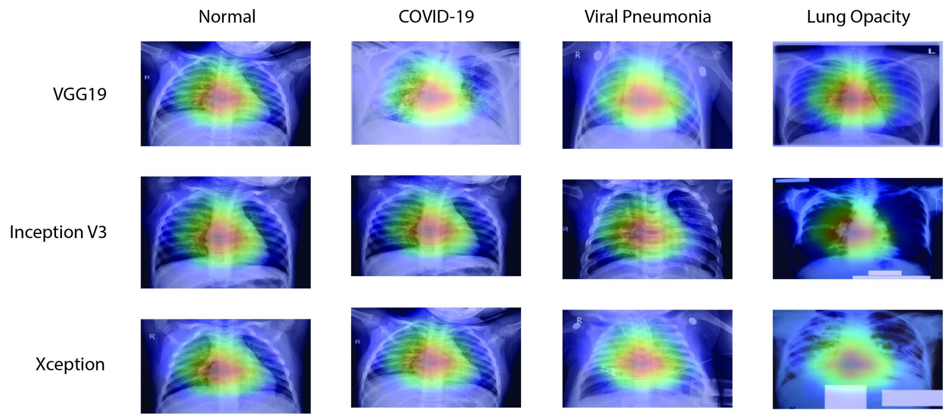


Figure 4.23: Superimposed images of the GradCAM Analysis at the last layer of the CNN models

In Figure 4.23, first row: four superimposed images of the last layer of the normal class, the COVID-19 class, the Viral Pneumonia class and the Lung Opacity class from the VGG19 model. The second row and the third row contain the superimposed images of the normal class, the COVID-19 class, the Viral Pneumonia class and the Lung Opacity class from the Inception V3 model and the Xception model respectively. In the same way, the first column contains the superimposed images of the normal class generated in the last layer of the VGG19 model, the Inception V3 model and the Xception model. The second column contains the superimposed images of the COVID-19 class generated in the last layer of the VGG19 model, the Inception V3 model and the Xception model. The third column and the fourth column contain the superimposed images of the Viral Pneumonia class and the Lung Opacity class generated by the same transfer learning models respectively.

Chapter 5

Conclusion

Three CNN-based transfer learning methodologies are presented in this paper. The three transfer learning methods are utilized to identify normal, COVID-19, lung opacity, and viral pneumonia pictures from X-ray images. Three methods are used to automatically recognize the normal class, the COVID-19 class, the Viral Pneumonia class and the Lung Opacity class from X-ray images. Three well-known and previously published CNN-based deep learning systems were adjusted using chest X-ray images to improve the ability to tell apart the images for the four class images. The picture sizes used for the models' training were 299 by 299 for the Xception model, 224 by 224 for the VGG19, and 224 by 224 for the Inception V3 model, in accordance with the model's standard criteria. Because this is a sensitive diagnosis for the healthcare industry, it is essential to train the model with an image size that does not lose any important information. The VGG19 model, the Inception V3 model, and the Xception model are each 0.9614, 0.9535, and 0.9262 percent accurate when given a normal image input. Applying explainable AI to the last layer of each of the three networks indicated that the VGG19 model was keeping an eye on the area of the chest where COVID-19, Viral Pneumonia and Lung Opacity have been located, indicating that it is the best of the three models. Additionally, the experiment shows that the final block of every model keeps track of information that is essential to classification, and the last block activation layer is in charge of spotting more complex characteristics since it is situated near to the model output. With respiratory failure being the main cause of mortality, COVID-19 has already presented a threat to the world's healthcare system. Additionally, Bangladesh is seeing an increase in viral pneumonia and lung opacity. The VGG19 model-based diagnosis described in this study has the potential to save lives through early screening and appropriate therapy. As doctors' time is limited due to the enormous number of patients seen outside of office hours or in an emergency, this diagnosis has the potential to reduce mortality rates. Our explainability may not only help doctors improve Covid-19 screening, but it may also act as a responsible and transparent audit of our models.

5.1 Challenges and Future Directions

Obtaining data is the most difficult endeavor due to the restricted and scarcity of healthcare data. To conduct a more thorough investigation and to create better models that can detect and diagnose illnesses at their very earliest stages, we believe that additional data is necessary. In the future, many anatomical illnesses may be the focus of an approach using 3D models. We envision a comprehensive system that can identify any sickness that affects individuals, greatly reducing their suffering.

Bibliography

[1] O. Ruuskanen, E. Lahti, L. C. Jennings, and D. R. Murdoch, "Viral pneumonia," *Lancet*, vol. 377, no. 9773, pp. 1264–1275, Apr. 2011, ISSN: 0140-6736. DOI: 10.1016/S0140-6736(10)61459-6.

[2] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative adversarial nets," *stat*, vol. 1050, p. 10, 2014.

[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[4] C. Frogner, C. Zhang, H. Mobahi, M. Araya, and T. A. Poggio, "Learning with a wasserstein loss," *Advances in neural information processing systems*, vol. 28, 2015.

[5] C. Szegedy, W. Liu, Y. Jia, *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[6] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International conference on machine learning*, PMLR, 2017, pp. 214–223.

[7] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[8] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," *Advances in neural information processing systems*, vol. 30, 2017.

[9] Z. Wan, Y. Zhang, and H. He, "Variational autoencoder based synthetic data generation for imbalanced learning," in *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2017, pp. 1–7. DOI: 10.1109/SSCI.2017.8285168.

[10] P. M. de Groot, C. C. Wu, B. W. Carter, and R. F. Munden, "The epidemiology of lung cancer," *Transl. Lung Cancer Res.*, vol. 7, no. 3, pp. 220–233, Jun. 2018.

[11] kmader, "Lung Opacity Overview," *Kaggle*, Oct. 2018. [Online]. Available: https://www.kaggle.com/code/kmader/lung-opacity-overview/notebook.

[12] A. Korotin, V. Egiazarian, A. Asadulaev, A. Safin, and E. Burnaev, "Wasserstein-2 generative networks," *arXiv preprint arXiv:1909.13082*, 2019.

[13] F. J. Moreno-Barea, J. M. Jerez, and L. Franco, "Improving classification accuracy using data augmentation on small data sets," *Expert Systems with Applications*, vol. 161, p. 113 696, 2020.

[14] L. Sun, J. Wang, Y. Huang, X. Ding, H. Greenspan, and J. Paisley, "An adversarial learning approach to medical image synthesis for lesion detection," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 8, pp. 2303–2314, 2020. DOI: 10.1109/JBHI.2020.2964016.

[15] A. Waheed, M. Goyal, D. Gupta, A. Khanna, F. Al-Turjman, and P. R. Pinheiro, "Covidgan: Data augmentation using auxiliary classifier gan for improved covid-19 detection," *IEEE Access*, vol. 8, pp. 91 916–91 923, 2020. DOI: 10.1109/ACCESS.2020.2994762.

[16] K. Ann, Y. Jang, H. Shim, and H.-J. Chang, "Multi-scale conditional generative adversarial network for small-sized lung nodules using class activation region influence maximization," *IEEE Access*, vol. 9, pp. 139 426–139 437, 2021. DOI: 10.1109/ACCESS.2021.3116034.

[17] A. Besbes, "Classify Knee injuries with PyTorch | DataDrivenInvestor," *Medium*, Dec. 2021. [Online]. Available: https://medium.datadriveninvestor.com/deep-learning-and-medical-imaging-how-to-provide-an-automatic-diagnosis-f0138ea824d.

[18] A. Chaubey, "Downsampling and Upsampling of Images — Demystifying the Theory," *Medium*, Dec. 2021. [Online]. Available: https://medium.com/analytics-vidhya/downsampling-and-upsampling-of-images-demystifying-the-theory-4ca7e21db24a.

[19] J. Han, M. R. Min, L. Han, L. E. Li, and X. Zhang, "Disentangled recurrent wasserstein autoencoder," *arXiv preprint arXiv:2101.07496*, 2021.

[20] Q. Li, Z. Wang, G. Li, J. Pang, and G. Xu, "Hilbert sinkhorn divergence for optimal transport," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3835–3844.

[21] S. Liu, E. Gibson, S. Grbic, *et al.*, *Manipulable object synthesis in 3d medical images with structured image decomposition*, US Patent 11,024,027, Jun. 2021.

[22] Z.-H. Liu, L.-B. Jiang, H.-L. Wei, L. Chen, and X.-H. Li, "Optimal transport-based deep domain adaptation approach for fault diagnosis of rotating machine," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, 2021. DOI: 10.1109/TIM.2021.3050173.

[23] Y. Wang, W. Sun, J. Jin, Z. Kong, X. Yue, *et al.*, "Wood: Wasserstein-based out-of-distribution detection," *arXiv preprint arXiv:2112.06384*, 2021.

[24] C. Zhao, R. Shuai, L. Ma, W. Liu, D. Hu, and M. Wu, "Dermoscopy image classification based on stylegan and densenet201," *IEEE Access*, vol. 9, pp. 8659–8679, 2021. DOI: 10.1109/ACCESS.2021.3049600.

[25] P. Janetzky, *Image Augmentations in TensorFlow - Towards Data Science*. Towards Data Science, Mar. 2022. [Online]. Available: https://towardsdatascience.com/image-augmentations-in-tensorflow-62967f59239d.

[26] *Loss Functions | Machine Learning | Google Developers*, [Online; accessed 12. Jan. 2023], Jul. 2022. [Online]. Available: https://developers.google.com/machine-learning/gan/loss.

[27] A. Sankar, *Demystified: Wasserstein GANs (WGAN) - Towards Data Science*. Poland: Towards Data Science, Jan. 2022. [Online]. Available: https://towardsdatascience.com/demystified-wasserstein-gans-wgan-f835324899f4.

[28] *The Discriminator | Machine Learning | Google Developers*, [Online; accessed 12. Jan. 2023], Jul. 2022. [Online]. Available: https://developers.google.com/machine-learning/gan/discriminator.

[29] *The Generator | Machine Learning | Google Developers*, [Online; accessed 12. Jan. 2023], Jul. 2022. [Online]. Available: https://developers.google.com/machine-learning/gan/generator.

[30] W. H. O. Who, "Pneumonia in children," *World Health Organization: WHO*, Nov. 2022. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/pneumonia.

[31] S. Yegulalp, "What is TensorFlow? The machine learning library explained," *InfoWorld*, Jun. 2022. [Online]. Available: https://www.infoworld.com/article/3278008/what-is-tensorflow-the-machine-learning-library-explained.html.

[32] *Chest X-ray (Covid-19 & Pneumonia)*, [Online; accessed 14. Jan. 2023], Jan. 2023. [Online]. Available: https://www.kaggle.com/datasets/prashant268/chest-xray-covid19-pneumonia.

[33] *Chest X-ray (Covid-19 & Pneumonia)*, [Online; accessed 17. Jan. 2023], Jan. 2023. [Online]. Available: https://www.kaggle.com/datasets/prashant268/chest-xray-covid19-pneumonia.

[34] *PyTorch Image Classification for Product Recognition: SOTA Framework for SKU Recognition | Width.ai*, [Online; accessed 9. Jan. 2023], Jan. 2023. [Online]. Available: https://www.width.ai/post/pytorch-image-classification.

[35] *WHO Coronavirus (COVID-19) Dashboard*, [Online; accessed 9. Jan. 2023], Jan. 2023. [Online]. Available: https://covid19.who.int.

[36] *Lung opacity: Understanding what this means*, https://www.healthline.com/health/lung-opacity, (Accessed on 01/09/2023).