Rare Disease Classification from Facial Photographs using Deep Learning

By

Hafsa Moontari Ali

A Thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE

Department of Computer Science
University of Manitoba

Abstract

Rare diseases affect a small number of populations all around the world. Often, rare disease patients are misdiagnosed and deprived of proper treatment due to the lack of knowledge about the diseases. The unavailability of standard data and methodologies to identify the rare diseases has made the situation more complex. Rare diseases are caused by malfunction of genes, and often leave noticeable traits on the face. In this thesis, a dataset of facial photographs of rare diseases is curated. The correlation between risk genes and facial features of rare diseases is calculated. Finally, rare diseases are classified from healthy facial photographs of children by employing transfer learning-guided pre-trained deep learning models. The performance of different convolutional neural network models (AlexNet, ResNet-18, ResNet-34, ResNet-50, VGG-16, VGG-19, DenseNet121, MobileNetV2) are analyzed over two different transfer learning approaches. All the models, except VGG-16, achieved superior results when trained with fine-tuned transfer learning approach than the other transfer learning approach where the convolution base was considered as a fixed feature extractor. The fine-tuned MobileNetV2 model showed the best classification result with 94.92% accuracy, and precision, recall, F1-score and AUC of 0.9498, 0.9492, 0.9475, 0.99, respectively. Then, augmentation is performed on ResNet-50 and DenseNet-121 and the overall performance improved in both transfer learning-based approaches. Two traditional machine learning based models (Support vector machine, eXtreme Gradient Boosting) are also applied for classification. The machine learning models achieved moderate results but underperformed comparing to the deep learning models.

Acknowledgement

First and foremost, I wish to express my sincere gratitude to my master's thesis supervisor Dr. Pingzhao Hu and co-supervisor Dr. Yang Wang for their unrelenting encouragement and mentoring. They continuously motivated me to achieve utmost expertise in research and writing of this thesis. Without their insightful reviews, I would hardly finish this thesis.

I would like to express my gratitude to Dr. Patrick Frosk for giving us his rare disease data table to use for research purpose. Thanks to Yolanda Ding, for curating the facial photographs of rare diseases as part of her summer internship in the Hu Lab. I would also thank to Rayhan Shikder, alumni of the Hu Lab to help me with the canonical correlation computation part of Chapter 4 in this thesis.

I would also like to extend my gratitude to my advising committee members Dr. Carson Kai-Sang Leung, and Dr. Max Turgeon for their valuable time and patience in reviewing my thesis work.

Special thanks to all my friends and colleagues at The Hu Lab. Their friendly discussions, weekly paper presentation followed by insightful discussions helped me in academic growth and kept me motivated during the process. Thanks to Dr. Hu for organizing the paper presentations and selecting the papers aligned with our ongoing research.

Thanks to all the faculty members and staffs from the Computer Science department, and the Biochemistry and Medical Genetics department who helped me in various steps of my degree.

Last but not the least I would like to express my gratitude to my family members for their encouragement and unconditional love for me to come at this stage and complete my degree.

Table of Contents..............4

List of Figures

List of Tables

Table of Abbreviations

Abbreviation	Description
RD	Rare Disease
AI	Artificial Intelligence
GO	Gene Ontology
DAG	Directed Acyclic Graph
CNN	Convolutional Neural Network
VGG	Visual Geometry Group
SVM	Support Vector Machine
XGBoost	eXtreme Gradient Boosting
k-NN	K Nearest Neighbor

List of Equations

Chapter 1: Background and Introduction

1.1 Rare Diseases

A disease is considered as being rare when it affects only a limited fraction of individuals from the general population comparing to other prevalent diseases. To date, nearly 7,000 rare diseases (RD) are found with < 5 cases per 10,000 population [1]. Even though the diseases are individually rare, almost 350 million people are affected by rare diseases worldwide [2]. Triple A syndrome, 3C syndrome, Alpha-mannosidase etc. are few examples of such diseases. The exact number of rare disease patients is difficult to estimate, as epidemiological data for many of these diseases are not available. Rare diseases are chronic, disabling and often life threatening. The heterogeneity in disease symptoms, wide range of disorders, limited data, geographic dispersion and unavailability in epidemiological data make rare diseases a challenging domain in research [3]. It creates crucial public-health issue and formidable challenges towards medical community. As a result, rare disease patients are often deprived of proper treatment and disregarded by the medical community and policy makers. But timely diagnosis can largely improve the patient recovery and life expectancy.

Rare diseases pose significant restrictions on the development of patient's physical, mental and social aspects. These degrade individual's quality of life, education and earning potentiality. An investigative assessment was performed on 2500 patients with chronic diseases to report their experiences in getting treatment [4]. Among them, 8.2% were rare disease patients. It was reported that they had worst experience in terms of treatment accessibility, social and economic loss. Their experiences were much more difficult than those patients not affected with rare diseases. Eurodis et al. addressed the diagnostic delay by doing a survey on patients from 17 European countries suffering from different rare diseases (Cystic fibrosis, Duchenne muscular dystrophy, Ehlers-Danlos syndrome, Marfan’s syndrome, Prader-Willi syndrome, tuberous sclerosis and fragile X syndrome) [5]. The authors found that, 25% of the patients were correctly diagnosed more than after 5-30 years when the symptoms first appeared. Moreover, before final diagnosis, 40% of them were diagnosed incorrectly. Incorrect diagnosis resulted into complicated consequences: 16% of the patients had to undergo surgery, 33% did not get proper treatment and 10% were assumed to have psychosomatic symptoms and were provided psychological aid. 25% of the patients had to travel in a different state/country to get a confirmed diagnosis.

1.2 Disease Identification from Facial Photographs

The relationship between human face and diseases has been investigated over a long time. In ancient times, the experienced doctors used to observe the facial features of patients to know about any lesions. The intuition was that the pathological changes of internal organs might be reflected in the face as changes in structure, form or morphology [6]. Modern medical researchers also indicate that many diseases leave noticeable features on human faces which can be greatly informative to clinical geneticists [7]. Such observation has led to disease screening, classification among multiple diseases, classification between disease and healthy faces etc. At present, the widespread usage of mobile phone, camera has made photo capturing quite easy. So facial appearance has become an important indicator in identifying various diseases and gradually paved the path for facial feature guided disease diagnosis.

The application of computer vision and image processing-based approaches in disease detection from facial photographs has been studied over 30 years. In these applications, various facial features extracted from facial photographs are given as input on which different analysis are performed. The existing studies [8][9] in the broader literature indicates that the facial feature extraction methods are divided into local feature based methods, holistic methods, statistical model based methods and deep learning based methods.

1.2.1 Traditional Methods

Previous research in local features-based methods showed that manual annotation of landmark features was used to extract disease specific facial patterns. Loos et al. [10] developed a computer based genetic syndrome recognition system which classified five diseases. They extracted 32 landmark features from face and used gabor wavelet transformation for preprocessing. Next, the facial feature vectors are calculated from face specific bunch graphs and diseases are categorized by recognizing facial resemblance. Vollmer et al. [11] extracted 48 landmarks, applied gabor wavelets along with the standardized coordinates of those landmarks and generated feature vectors with geometric and textural features. These feature vectors are later analyzed to classify diseases by detecting certain facial patterns. The authors here examined the classification accuracy of the proposed method by increasing the number of syndromes from 10 to 14. Cornejo et al. [12] evaluated the appearance based local features in distinguishing genetic disorders from facial photos. They used Oriented FAST (Features from Accelerated Segment Test) and Rotated BRIEF (Binary Robust Independent Elementary Features) - ORB and fused it with geometric features (coordinates and distance of landmark points).

Statistical model-based methods depend on the principles of statistical analysis of shapes. Ferry et al. [13] applied a statistical model, Active Appearance Model (AAM) to learn how the shape and texture vary across the facial images. The authors also used the AAM model to visualize the canonical phenotypes from faces.

Holistic feature-based methods extract global information from the entire face. The global information is represented by small number of features which capture the variance among individual faces. Eigenface [14], based on Principal Component Analysis (PCA) is a commonly used holistic method in facial analysis. Another holistic method is Fisherface [15], which is based on Linear Discriminant Analysis.

1.2.2 Artificial Intelligence based Methods

With the advent of artificial intelligence (AI), machine learning and deep learning-based methods are shining brilliantly in image analysis. The aim of machine learning is to learn specific patterns and relationships from examples and observations for solving complex problems. There are various kinds of machine learning models, including artificial neural networks, decision trees, support vector machines, regression analysis, Bayesian networks, and genetic algorithms [16]. Deep learning is a model of machine learning based on artificial neural network. It consists of multiple hidden layers and therefore, these neural networks are referred as deep neural network and the framework is called "deep learning". Deep learning models learn conceptual abstraction from data in both hierarchical and composite way, uncovering many underlying characteristics ignored by human beings. The higher levels of abstraction are built on top of the low levels and the lower levels are the input data (e.g. facial image). Deep learning makes use of many nonlinear functions to model the dependency between the features and labels. The success of deep learning depends on the usage of graphical processing unit (GPU) and the availability of large datasets containing millions of training data. Besides, the development of open source software platforms like Pytorch [17], Caffe [18], Theano [19], and Tensorflow [20] has made it easy to train deep networks and reuse the latest models.

Several high impact factor journals and top-tier conference publications have established that the AI based models are playing an increasing role in medical research and clinical practice in assisting disease screening, diagnosis, prediction of response to therapy and many more [21] [22][23]. Zhao et al. detected down syndrome from facial photographs by employing Support Vector Machine (SVM) with radial basis function and performed classification by analyzing the geometric and textural features [24]. Kruszka et al. [25] also applied SVM to detect 22q11.2 DS syndrome from facial photographs by extracted 126 facial features from geometric and texture biomarkers.

Acromegaly is a rare hormonal disorder caused by excessive amount growth hormone (GH) produced in the body. This disease changes face by enlarging nose, jaw, forehead, and cheekbone, thickening the skin and increasing the space in tooth [26]. Schneider et al. [27] developed a face classification software which acquired superior results than medical experts and general internists in detecting acromegaly from facial photographs (both frontal and side view). Kong et al. [28] classified the acromegaly patients and healthy controls from facial photographs by employing several machine learning based models (Linear Model, k-Nearest Neighbor, Support Vector Machine, Random Tree). The authors applied each of the models separately on the dataset and later used the ensemble method, bagging to combine outputs from the different models. The ensemble method achieved better result that the machine learning models and outperformed the specialists and primary care doctors.

The advancement of transfer learning has guided the researchers to use pre-trained model in disease diagnosis and classification. Jin et al. [29] performed facial diagnosis of four diseases:

Beta-thalassemia, Hyperthyroidism, Down syndrome and Leprosy. The characteristics of these diseases on facial surface are bone deformities, small eye opening, short nose, thin upper lip, underdeveloped jaws, staring eyes, thinning eyes, less blinking, nervousness, upward slanting eyes, small/flattened nose, small mouth, pale skin, eye damage and facial disfigurement. The authors utilized these distinguishing facial features in both single and multi disease classification. The pre-defined deep learning architectures with transfer learning approaches are used here.

Gurovich et al. [30] developed a facial image analysis framework, DeepGestalt by employing Deep Convolutional Neural Network (DCNN) to classify genetic syndromes. DeepGestalt learnt facial representation from Casia-WebFace dataset and applied knowledge transfer in genetic syndrome domain through fine-tuning. Shukla et al. [31] used the AlexNet model trained on ImageNet dataset and added SVM on the last layer of the architecture to classify six genetic syndromes from facial photographs. Singh et al. [32] applied a ResNet50 architecture, pre-trained on VGGFace to classify rare genetic diseases. They added three fully connected layers on top of the ResNet50 architecture and only retrained the last fully connected layers treating the remaining architecture as a fixed feature extractor.

The literature review shows that although there has been research on facial feature guided face recognition or disease identification, there is no well explored study to classify rare diseases from facial photographs of children. There is no publicly available facial photograph dataset of children with rare diseases. We want to fil this gap and establish such a dataset from publicly available resources. Next, we aim to perform classification analysis on the dataset employing convolutional neural network-based models. The performance of such models on rare disease facial photograph analysis will be extensively evaluated and analyzed.

1.3 Genetic Analysis

Rare diseases have genetic etiology which influences the treatment responses of different genetic mutation carriers. Over the past few decades, researchers have made advancement in the study of complex and rare human genetic disorders. It has become possible because of the collective knowledge of human genome sequence and the ability of advanced technologies to correctly associate the disease phenotype. Each cell in the human body consists of 23 pairs of chromosomes and each chromosome is made up of about 2,000 genes. Sometimes, any of the gene or protein can be absent, mutated or an extra chromosome can be present.

Alpha-mannosidosis [33], an inherited lysosomal storage disorder is caused by mutations in the MAN2B1 gene located on chromosome 19. This disorder affects around 1 in 500,000 live births. The characteristics of this rare disease include immune deficiency, skeletal abnormalities, hearing impairment, mental function and speech impairment. Van et al. investigated that Triple A syndrome is caused by mutations in the AAAS located at chromosome 12q13 [34]. Another rare disease, Sialidosis type 2 is caused by the biallelic mutation in NEU1 gene [35]. These are only few diseases and corresponding risk genes among the 8,000 rare diseases scattered around the world.

The biological vocabulary, Gene Ontology (GO) describes the functions of genes and acts as a foundation of genetic computational analysis. The degree of relatedness between the genes can be measured by the similarity based on their annotation. The idea behind the semantic similarity measurement is that genes with similar function are supposed to have similar annotation vocabulary and a close relationship in ontology structure. Semantic similarity validates the outcomes from biomedical studies such as gene clustering, gene expression data analysis and disease gene prioritization. Several studies on semantic similarity analysis have been published in the literature evaluating diverse approaches [36] [37] [38] [39]. Information content (IC) based methods [36] - [39] and graph based method [40] are two major categories of semantic similarity analysis approaches.

1.3.1 Genotype-Phenotype Relationship

Facial phenotypes of rare disease patients and the responsible risk genes both have potentiality in analyzing rare diseases. Both possess high informative features to facilitate the understanding of rare disease characteristics. Spiga et al. [41] collected facial images of 200 Alkaptonuria patients and developed an integrated platform to store their genetic, biochemical, histopathologic, clinical data and images. The authors applied k-mean algorithm and hierarchical clustering based on the collected genotypic and phenotypic data to perform stratification on the patients in order to create subgroups with similar features.

Little research has been conducted to investigate whether both the features extracted from the same set of data have any underlying relationship. So, it is worth investigating if the genotypic and phenotypic features are correlated or not. The examination and interpretation of the relationships between these two types of features may accelerate the disease detection approaches. The earlier a rare disease is identified, the more likely it is to get rapidly diagnosed and covered by effective medical treatment. It may also contribute notably to drug repurposing.

Facial feature guided disease diagnosis and genotypic-phenotypic analysis have been investigated over a long period of time. But very little research has been conducted on the rare disease patients, especially children. Moreover, that research only considered a few rare diseases leaving behind the vast number of rare diseases from worldwide. In this study, we address this gap and aim to curate facial photographs of rare diseases and keep the photographs in a database which will act as a data source for future research. Our next objective is to perform correlation analysis between the risk gene and facial phenotypes of the curated photographs and final objective is to classify rare disease facial photographs from healthy faces.

Chapter 2: Motivation and Research Objectives

2.1 Motivation

The rare diseases are heterogeneous in nature and the symptoms are often hidden. Rare disease data demands efficient analysis to get fruitful insights for disease identification and diagnosis. There is a lack of standardized rare disease data to identify such diseases. So, a standardized collection of data is required for research purpose.

The unavailability of rare disease dataset is crucial challenge in research community. We find it very important and aim to collect facial photographs of children affected with rare diseases. We focus on only children because the earlier a rare disease is identified, sooner treatment can be started. But these diseases are rare, and it is difficult to collect a large amount of data. Besides, rare disease-specific patients are geographically scattered, and correct information is not well-documented. In addition, there might be personal and ethical issue from the patient and family to disclose and use the information. These direct us to search for facial photographs of rare disease patients in publicly available sources and use the photos in our study. The development of a database is also necessary to store the photographs so that in future more data can be added in there.

The correlation between genetic and facial feature of rare disease children is significant in clinical and medical research. Based on the correlation between rare disease specific genes and corresponding facial features, the photographs can be considered as a key indicative of that rare disease. It can be considered as a characteristic face for unknown or ultra rare disease identification research.

Facial photographs of rare diseases need to be correctly classified from normal facial photographs. We assume that the curated facial photographs will be relatively small, so we limit our task in classifying rare diseases and healthy children based on their facial photographs. So, it will be a binary classification problem. Correct classification of diseases is a pre-requisite of fast disease diagnosis. Such a system with an acceptable error rate can significantly help people to take medical examinations in underdeveloped areas where expert doctors are not available or unable to identify rare diseases.

2.2 Research Objectives

In this study, 480 facial photographs of 104 rare diseases are collected from publicly available resources. This number is indeed very small. It is not possible to classify all the rare diseases from such small number of facial photos. So, all the 104 rare diseases are considered one class and facial photographs of healthy children (collected from two publicly available datasets) are considered as another class. Binary classification is performed to classify rare disease facial photos from healthy facial photos. Additionally, the correlation score is computed between the facial phenotype and gene similarity features from the curated facial photographs and the responsible risk genes. The research objective of this thesis are as follows:

1. Collect a list of rare diseases, facial photographs and implement a database to store the collected information.

2. Analyze the genotype-phenotype association between the genes and corresponding facial features employing Canonical Correlation Analysis.

3. Develop deep learning-based models for classification of rare diseases from the collected facial photographs.

Chapter 3: Data Collection and Database Construction

3.1 Introduction

In every discipline of research, accurate and honest collection of data is the key and primary step. Proper pre-processing, analysis, and measurement of data leads towards meaningful information essential for further processing or to make an informed decision. The success of evaluating novel methodologies or the performance analysis of existing approaches greatly relies on data. Moreover, in complicated cases like rare disease research, proper data collection is of utmost importance. Since the rare disease-affected children are geometrically scattered and disease symptoms cover a wide range of variations, it is difficult to collect many facial photographs. There exists some ethical and privacy issue also. The patient or their family may not feel comfortable disclosing their photographs or information. As a result, the facial photographs used in this thesis are collected from publicly available sources.

In addition to data collection, data storage and management is equally important. There are several advantages of storing data in a database, such as large amount of data can be stored, data can be quickly found by searching or sorting, and more data can be added to the database in future. Besides, storing the data in a database ensures security, maintains and views the relationship among different entities, and makes it easy to conduct future research on the data. A database system is implemented to store the collected rare disease photographs.

This chapter discusses the data collection steps and database construction.

3.2 Rare Disease Data Collection

3.2.1 Baseline Rare Disease List

The rare diseases used in this thesis were curated based on the data table from Dr. Patrick Frosk. He is a Clinician Geneticist and Associate Professor at the University of Manitoba. His collection has a focus on rare diseases localized or more prominent in Manitoba. The rationale behind considering Manitoba as location is that the patient population is ethnically quite diverse here and contains several founder populations. These populations, due to their unique structure, often have specific deleterious alleles at much higher frequency than seen in the general population and therefore have overrepresentation of specific monogenic disorders. Since many of these conditions are ultrarare anywhere else in the world, investigating and diagnosing patients from such populations requires detailed knowledge of these conditions and how they might present in clinical practice. To facilitate clinical care as well as aid the work of local genetic researchers, a listing of conditions and their causative genetic variants in these populations was created. Creation of the list included a thorough review of the literature as well as review of the local clinical databases, both at the level of the clinic itself as well as the only clinical genetic testing laboratory in the province. This list is a living document and is regularly updated as new diseases and/or genetic variants are identified. In this thesis, this list is considered as a baseline for rare disease data collection.

3.2.2 Modified Rare Disease List

The baseline RD list is modified to fit the objective of this research. Dr. Patrick Frosk's list focused on population type (Indigenous groups) and genetic inheritance (**Table 3.1**). But data collection will be limited if the indigenous population is considered as a searching parameter. Besides, the inheritance type in Dr. Frosk's data is classified in three categories: autosomal recessive (AR), autosomal dominant (AD) and X-linked (XL). This information is too broad to use as a criterion in this thesis's data collection.

Table 3.1: Portion of Dr. Patrick Frosk's data table

Disease	Gene	Mutations	Disease Type	Population
Alstrom	ALSM1	p.Q3494X	AR	Mennonite
Aicardi - Goutieres	TREX1	p.R164X	AR	First Nations
…	…	…	…	…
Ankyloglossia cleft palate	AR	p.L676P	XL	Hutterite
Bowen Conradi	EMG1	p.D86G	AR	Hutterite

So, the columns "Disease Type" and "Population" are not considered and any overlapping diseases from different populations are removed as well. The modified data table of Dr. Frosk used in this thesis is shown in Table 3.2.

Table 3.2: Portion of modified data table for thesis

Disease	Verification	Disease Abbreviation	Gene	Mutations	# of photos	Photo 1	Photo 2	Photo 3	Photo 4	Photo 5
Alstrom	yes	ALS	CYP17A1	p.Q3494X	5	Link	Link	Link	Link	Link
Aicardi-Goutieres	yes	AIC	TREX1	p.R164X	5	Link	Link	Link	Link	Link
…	…	…	…	…	…	…	…	…	…	…
Ankyloglossia cleft palate	yes	ANK	AR	p.L676P	2	Link	Link	-	-	-
Bowen Conradi	yes	BOW	EMG1	p.D86G	0	-	-	-	-	-

3.2.3 Verification of the Selected Rare Diseases in Orphanet

The names of the rare diseases were checked with Orphanet database [42] to verify whether Dr. Frosk's list are officially rare diseases. Orphanet is a widely used portal for rare diseases and orphan drugs. It provides high-quality information of rare diseases with causing genes, directory of patient organization, expert centers, medical laboratories and ongoing research. The diseases of the modified RD list are manually verified whether they are present in the Orphanet. This information added in the modified table (Second column of **Table 3.2**). Only the verified diseases are considered as reference to curate the facial photos. For each disease, disease abbreviation is generated considering the first three or four characters of the disease name and documented in the table (Third column of **Table 3.2**). The disease abbreviations are treated as unique information for each disease.

3.2.4 Rare Disease Photo Curation

The goal is to curate the facial photographs for each rare disease listed in **Table 3.2**. Since the diseases are rare and publishing facial photographs publicly has potential ethical issues, it is presumable that it is not possible to get many photographs. A goal is set to curate at least 1 – 5 photos for each disease (preferably 5). Some rules are set during photo curation such as: frontal-facing, portrait-style, single patient, open eyes and decent quality. The photos that do not satisfy the rules are either disregarded or manually altered to fit in these criterions. Only the facial photos of children are considered because the majority of the affected are children. Besides, in case of adults, multiple diseases may exist which can possibly make the rare disease symptoms less revealing.

The facial photos were curated by searching the disease name and some keywords like “face”, “child”, “patient” on Google Images. Among the search results, facial photos were collected from published journals and hospital foundations. In some cases, news articles, rare disease blogs and awareness campaigns were also considered as sources. The photos were considered after reviewing whether the sources are valid and other criterions are satisfied. The number of collected photos for each rare disease is listed in sixth column of **Table 3.2**. The sources of the photos are included from columns 7 – 11.

Only 60% of the collected diseases have five photos. Therefore, another database called Human Phenotype Ontology (HPO) was explored to determine the alternatives for the remaining diseases. HPO database provides hierarchical classification to find a disease subcategory [43]. The target was to find a replacement closest to the original rare disease. The disease for which photos are not found, it is searched in HPO. The closest disease was selected from there and verified whether five photos were available for it. If not, search was continued through another branch. The replacement diseases were verified on Orphanet. Once all the diseases and photos were collected, the final dataset consists of 104 diseases, 480 photos, and 88 diseases with a complete set of 5 photos. The summary of the dataset is presented in **Table 3.3**.

Table 3.3: Summary of the dataset

Type	Number
Total Rare Diseases	104
Total Facial Photographs	480
Rare Disease Subcategories	71
RDs with 5 photographs	88
RDs with 4 photographs	3
RDs with 3 photographs	3
RDs with 2 photographs	7
RDs with 1 photograph	5

3.3 Database Construction

3.3.1 Database Design and Interface

A browser-based database system is developed, and the rare disease images are stored in there. The database is available here: http://10.246.3.157/. This database can be accessed after activating UofM VPN.

The major goal of this database construction is as follows:

(i) Organize data in a flexible way for easy and fast access.

(ii) Store, handle and manipulate data by means of add, browse or search functionalities.

The functionalities of the developed database include adding rare disease records, browsing the database and searching for diseases. Each disease record consists of disease name, unique ID, disease abbreviation, disease subcategory, one or multiple risk genes, reference of the risk gene(s) and some facial images. The unique ID of rare disease is ORPHA code, which is a unique numerical identifier generated by the Orphanet website. The interface of the database is designed considering the input data. The interface is kept simple, and the functionalities (Search, Browse, Add Disease) are added on top for visibility (**Figure 3.1**).

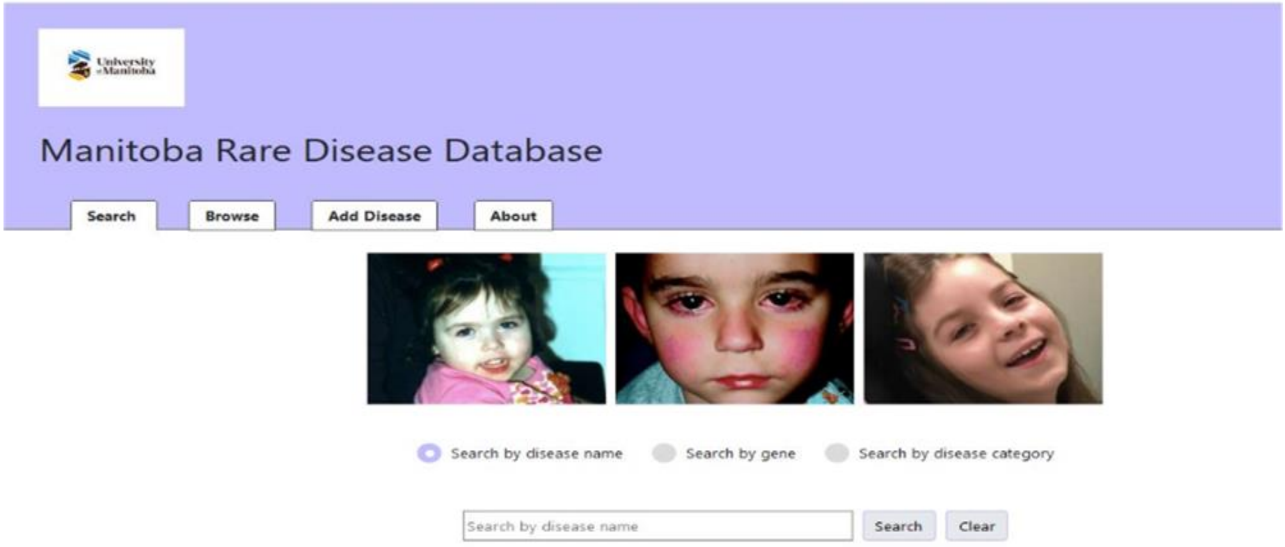
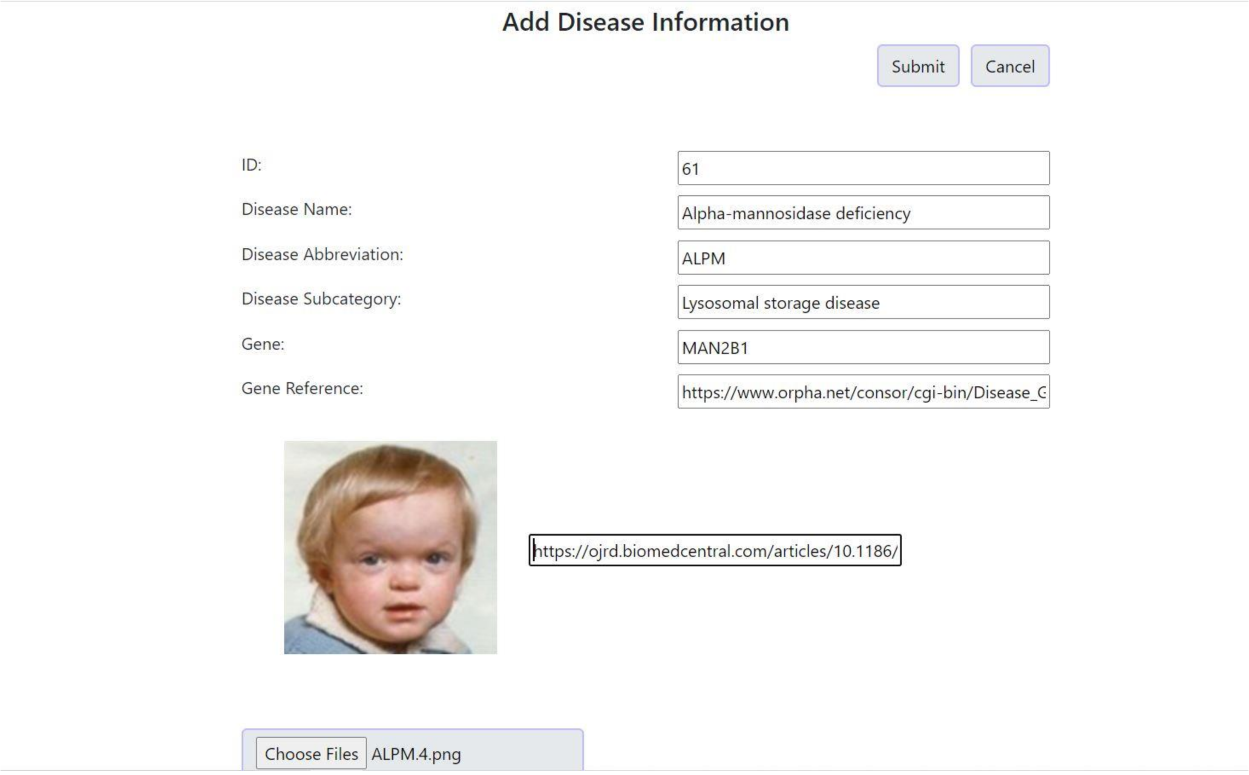


Figure 3.1 : The interface of the RD database

In order to store data, the disease records need to be added in the database. This is done by clicking on "Add Disease" functionality. The disease record will be inserted by providing inputs to the textboxes and images will be added by clicking on "Browse". Finally, the task will be completed by clicking on "Submit" in the top right corner of the page. **Figure 3.2** presents how a disease record is added in the database.



Add Disease Information

ID:	61
Disease Name:	Alpha-mannosidase deficiency
Disease Abbreviation:	ALPM
Disease Subcategory:	Lysosomal storage disease
Gene:	MAN2B1
Gene Reference:	https://www.orpha.net/consor/cgi-bin/Disease_G

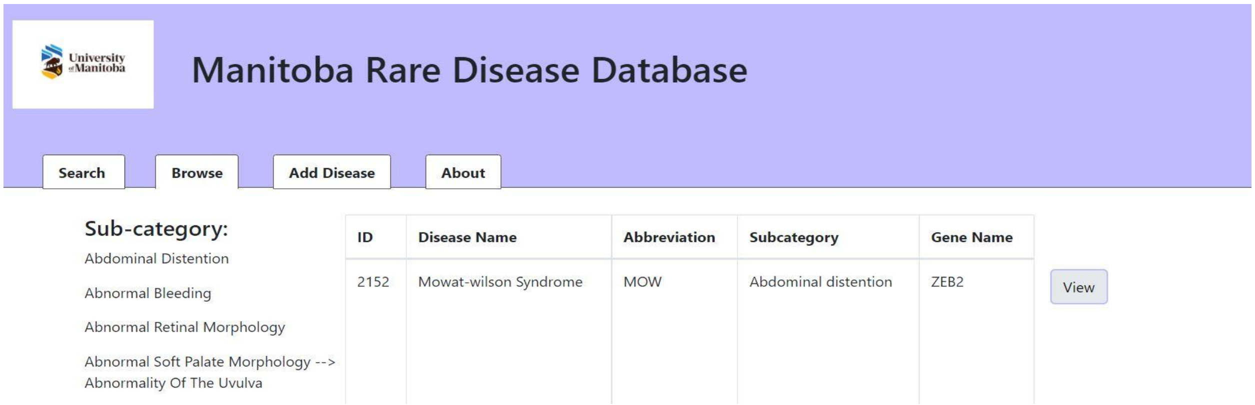
https://ojrd.biomedcentral.com/articles/10.1186/

ALPM.4.png

Figure 3.2 : Disease record is inserted in the database

The disease names and ID are the mandatory fields. Any other fields except these two can be left blank (if information is unavailable) and the disease record will be added to the database.

The "Browse" functionality enables the user to view the stored data records in terms of subcategory. **Figure 3.3** shows that the disease Mowat-wilson Syndrome is displayed after selecting the Abdominal Distention subcategory.



Sub-category:

Abdominal Distention

Abnormal Bleeding

Abnormal Retinal Morphology

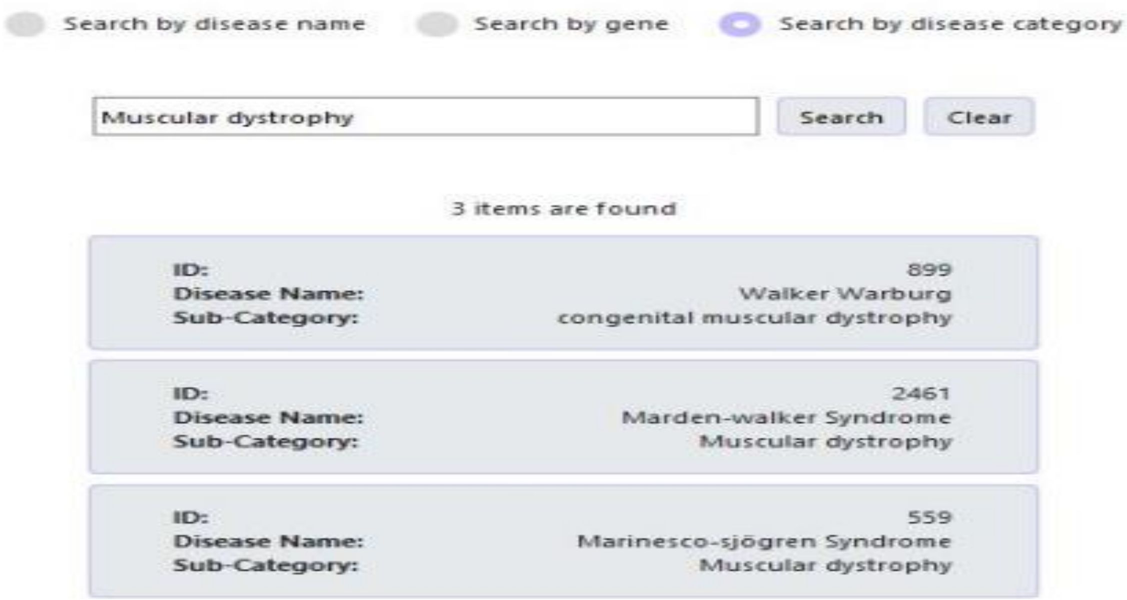
Abnormal Soft Palate Morphology --> Abnormality Of The Uvulva

ID	Disease Name	Abbreviation	Subcategory	Gene Name
2152	Mowat-wilson Syndrome	MOW	Abdominal distention	ZEB2

Figure 3.3 : Browse by disease subcategory

The disease records can be edited by accessing the "View" option in the "Browse" page. The edit option is necessary because any wrong information might be inserted or more images of a rare disease may be found later which need to be stored.

The search function operates in multiple ways: search by disease name, by gene or by disease subcategory. If the searched record exists in the database, it will be displayed in the page including associated information. Otherwise, "No record found" will be displayed. Even though each rare disease has different name, multiple diseases may belong to the same disease subcategory. In such case, if a user searches by disease subcategory, all the diseases from that subcategory will be displayed. The search by disease subcategory option is depicted in Figure 3.4. Search is performed here by disease subcategory, Muscular dystrophy. This subcategory has three diseases, and all three are displayed.



○ Search by disease name ○ Search by gene ● Search by disease category

Muscular dystrophy

3 items are found

ID:	899
Disease Name:	Walker Warburg
Sub-Category:	congenital muscular dystrophy
ID:	2461
Disease Name:	Marden-walker Syndrome
Sub-Category:	Muscular dystrophy
ID:	559
Disease Name:	Marinesco-sjögren Syndrome
Sub-Category:	Muscular dystrophy

Figure 3.4: Search by disease subcategory

3.3.2 Implementation of the Database

The database was implemented using MySQL database system in the back-end. In the front-end, the pages were created using HTML and Javascript. The python language was used with flask framework to build a REST API. When the user interacts with the system (search / browse / Add disease information), mysql queries are operated on the dataset. Based on the functionalities, the used queries are: SELECT, DELETE, INSERT and UPDATE. Finally, the front end fetches the resultant records from the MySQL database through REST API.

3.4 Conclusion and Discussion

The database is a much more efficient mechanism than spreadsheets to store and organize data. A web-based database provides a centralized facility to access, modify or use data from anywhere in the world. In this study, a total of 480 facial photographs of rare disease children are stored in the implemented database. The number of photographs is relatively small. In future, more facial photographs will be collected and stored to get insightful directions from data analysis.

At present, the database provides the features for data storage and manipulation. It is only accessible through the UofM network. Further enhancements can be done on the database for greater control of rare disease information. One important enhancement is to make the database publicly available and add features to handle user privacy and authentication. Under the “Browse” functionality, all the 71 disease subcategories are shown in a single page. If more disease subcategories are added in future, placing all these in a single page will be unfeasible. So, pagination should be added so that the subcategories are placed in different pages. The user interface design of the database can be improved.

Chapter 4: Genotype-Phenotype Correlation Analysis

4.1 Gene Similarity Estimation

Rare diseases are caused by altered functions of genes. Different kinds of genes can be responsible for one rare disease. For simplicity, only the risk genes are considered as responsible in this thesis. As depicted in **Table 3.3**, a total of 104 rare diseases are found during data collection. Among these 104 diseases, 78 diseases are monogenic and 8 of the diseases have multiple risk genes. Risk gene information is not found for remaining 18 rare diseases, from where the data was collected. So, these 18 rare diseases are excluded from consideration.

The pair-wise semantic similarity of the 86 risk genes was computed by applying a graph based method [40]. Based on this method, the semantic similarity is computed considering the locations of the genes in the graph and their relationship with the ancestor terms.

An ontology is a formal representation of a body of knowledge within a given domain. The gene ontology (GO) is a framework designed to describe the function of genes [44]. It represents the domain knowledge in such a way that the computational representation of biological systems can be supported. It is a structured and controlled vocabulary of terms. The terms are subdivided in three ontologies: Molecular Function (MF), Biological Process (BP) and Cellular Component (CC), each describing a specific aspect of gene functionality. GO annotations describe the association of gene with GO terms. Hence, GO annotations capture statements about the molecular actions of gene products, the biological processes it executes and the cells where it functions.

Semantic similarity computes the closeness/similarity between two GO terms and returns a numeric value representing the closeness. The GO terms are presented as nodes in directed acyclic graph (DAG) and two kinds of semantic relations (‘is-a’: class-subclass relation and ‘part-of’: partial ownership relation) form edges among the nodes. Each edge is assigned a weight based on the type of relationship. This DAG structure lays the foundation for quantitative semantic comparison among the terms. This type of hierarchical structure provides flexibility in semantic similarity computation because, if a gene is associated with a term, it is also associated with all the parents of that term [45]. And in GO context, one gene is usually annotated by multiple GO terms.

In the graph-based method, the similarity between the biological processes of two genes is determined by comparing the semantic similarities of GO terms. The semantic similarity of a GO term is determined using the topology of GO graph structure. For example, 0.8 and 0.6 can be the weights for ‘is-a’ and ‘part-of’ relations, respectively. For a term A_1 and its ancestor A_2, the semantic contribution of A_2 to A_1 is defined as the product of all edge weights in the best path from A_2 to A_1. The GO term is presented as **Equation 4.1**,

$$DAG_A = (A, T_A, E_A) \tag{4.1}$$

where A is the GO term, T_A and E_A are the set of GO terms and the set of edges connecting the GO terms in DAG_A, respectively.

The contribution of a GO term *t* to the semantics of GO term *A* is interpreted as the *S*-value of GO term *t* related to term *A*. For any term *t* in $DAG_A = (A, T_A, E_A)$, the *S*-value related to term A, is defined as **Equation 4.2**:

$$S_A(t) = \max\{w_e \ * S_A(t') \mid t' \ \in childrenof\ (t)\}, t\ \neq A \tag{4.2}$$

The semantic value of GO term A, $SV(A)$ is computed as **Equation 4.3**:

$$SV(A) = \sum_{t \in T_A} S_A(t) \tag{4.3}$$

The semantic similarity between two terms is calculated following two steps:

1. The sum of semantic contributions of all common ancestors to each of the terms is computed.
2. The total semantic contribution of each term's ancestors is divided by that term.

So, the semantic similarity value of two GO terms A and B is defined as **Equation 4.4**:

$$S_{GO}(A, B) = \frac{\sum_{t \in T_A \cap T_B}(S_A(t) + S_B(t))}{SV(A) \ + SV(B)} \tag{4.4}$$

If the semantic similarity value between two genes lies in the high end of the range from 0 to 1, the genes are considered analogous in terms of their biological processes. The GOSemSim-package of R is used to calculate the semantic similarity [46]. This package identified the semantic similarity relationship of 77 genes among the 86 genes. As a result, a 77 × 77 matrix is generated.

Table 4.1: A portion of the 77 × 77 gene semantic similarity analysis matrix

	AAAS	MAN2B1	BBS2	…	LONP1	SLC39A8	DPH1
AAAS	1	0.268	0.216	…	0.307	0.129	0.495
MAN2B1	0.268	1	0.135	…	0.288	0.11	0.629
BBS2	0.216	0.135	1	…	0.144	0.143	0.25
…	…	…	…	…	…	…	…
LONP1	0.307	0.288	0.144	…	1	0.068	0.38
SLC39A8	0.129	0.11	0.143	…	0.068	1	0.162
DPH1	0.495	0.629	0.25	…	0.38	0.162	1

4.2 Facial Phenotype Extraction

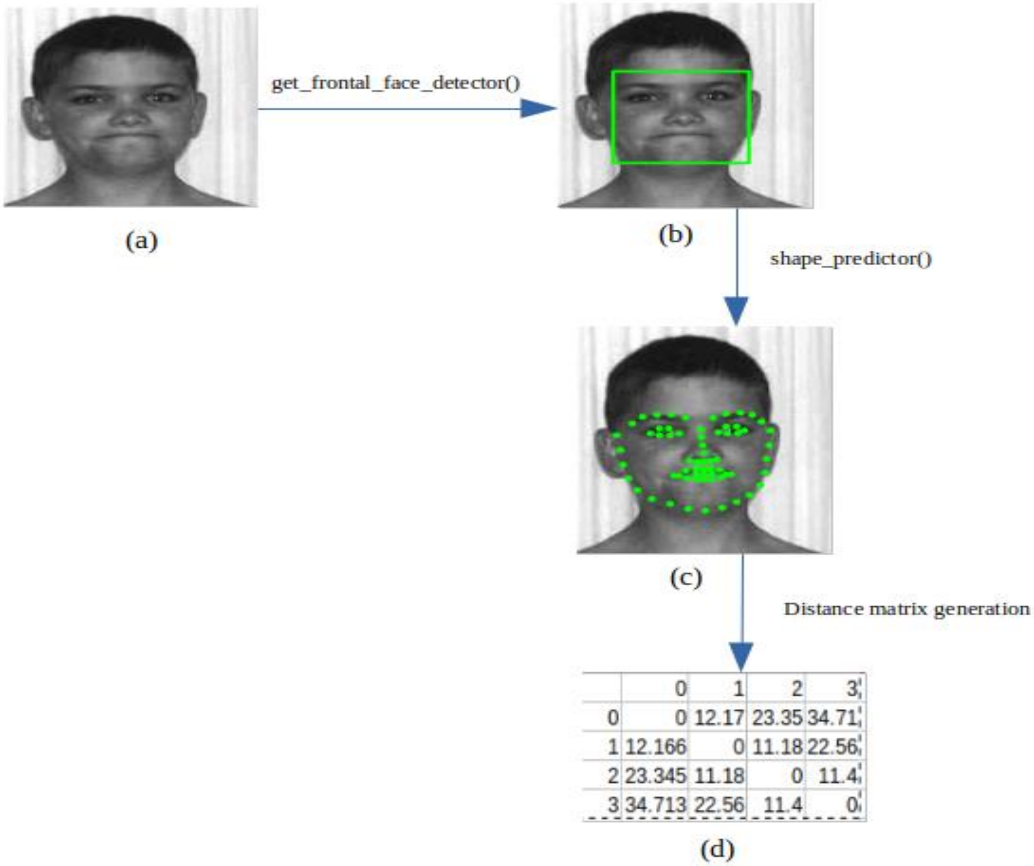
In this step, facial landmarks from the rare disease patient photographs are detected and the geometric distances are calculated. The dlib library in python is used for facial landmark detection [47]. It is an advanced machine learning library frequently used to solve computer vision problems. It is written in C++ and it works with C/C++, python and Java. The steps of facial phenotype extraction is briefly described below and summarized in **Figure 4.** 1.

4.2.1 Face Localization

The facial photographs are resized to 224×224 by facial frontalization is performed using the dlib library. The faces from the photographs are first localized using the get_frontal_face_detector() function in dlib library. It returns a function containing the points where the face is specifically located and renders a rectangle over the face. Among the 480 curated facial photographs, faces regions are successfully located from 427 images.

4.2.2 Facial Landmarks Detection

Total 68 facial features like eyes, mouth, jaw points, nose points etc. are detected from the localized face region derived in the previous step. The Dlib library applies shape_predictor() function and generates a map of 68 points surrounding the features [48]. The shape_predictor() uses a manually labelled pre-trained model for landmarks detection. It applies an ensemble of regression trees to estimate facial landmarks from the input image.



	0	1	2	3
0	0	12.17	23.35	34.71
1	12.166	0	11.18	22.56
2	23.345	11.18	0	11.4
3	34.713	22.56	11.4	0

Figure 4. 1: The workflow of facial phenotype extraction. (a) The input facial image (b) The face region is located from the input (c) Facial landmarks are identified from the located region (d) A portion of the calculated 68 × 68 distance matrix for the given image

4.2.3 Distance Matrix Generation

Euclidean distance is calculated from each of the detected landmark points to all other points. Thereby, total 427 distance matrices are generated with shape of 68 × 68. Total 427 distance matrices are generated because among the 480 images, face is localized in 427 images. For the remaining images, face is not localized and therefore, those images are not considered for the next steps.

4.2.4 Distance Matrix to Disease Similarity Matrix Generation

The 68 × 68 matrices are converted into a 68 × 1 vector taking the average for each row. It is repeated for all the 427 matrices and finally a 68 × 427 matrix is formed. Among the 427 vectors, average is computed from the ones belonging to the same diseases. As a result, the facial image similarity matrix is converted into a disease type matrix. The resulting matrix is of shape 68 × 104.

Finally, those diseases are discarded from the 68×104 matrix for which risk gene information is not found. So, it resulted into a 68×77 disease similarity matrix. After transpose, it will become 77×68 matrix.

The steps are illustrated in PSEUDOCODE 1.

PSEUDOCODE 1

Facial image similarity matrix to disease matrix

```
Input: facial image dataset $F = \{D_{input}\}_{k=1}^{480}$
    $number\ of\ similar\ column\ names\ n$
Output: disease matrix $D_{68\times104}$
  repeat
    for all $D_{input} \in F$ do
      $D_{detected\_faces} \leftarrow detector\ (D_{input})$
      $D_{landmarks} \leftarrow predictor(D_{detected\_faces})$
        repeat
         for all $i \in (0, 68)\ do$
           for all $j \in (0, 68)\ do$
             $D_{68\times68} = Euclidean\_Dist(D_{landmarks[i,j]})$
         $D_{68\times1} = mean(D_{68\times68})$
```

$$D_{68\times427} = append(D_{68\times1})$$

$$D_{68\times104} = mean(D_{68\times427}, n)$$

4.3 Canonical Correlation Analysis

Canonical correlation analysis (CCA) measures the associations between two sets of feature matrices. In this study, the two feature matrices are generated in Subsection **4.1** and Subsection **4.2**. It finds the linear combinations of features from two datasets (X_1 and X_2) which are maximally correlated [49]. Here, X_1 and X_2 represent the gene semantic similarity matrix and disease similarity matrix, respectively.

In this section, the performance of two popular canonical correlation analysis methods is analyzed such as: regularized canonical correlation (RCC) analysis and sparse canonical correlation analysis (SCCA). The methods are evaluated in terms of correlation score. In the next sections, the methods are discussed.

4.3.1 Canonical Correlation Analysis (CCA)

CCA determines the linear combinations of the features from two feature matrices, X_1 and X_2 with $(n \times p_1)$ and $(n \times p_2)$ dimensions on the same subject $i = 1,2,3, \dots., n$. The goal of CCA is to maximum the **Equation 4.5**:

$$corr(w_1^T X_1, w_2^T X_2)$$

$$\text{or} \tag{4.5}$$

$$\left(\frac{w_1^T \sum_{12} w_2}{\sqrt{w_1^T \sum_{11} w_1 w_2^T \sum_{22} w_2}}\right)$$

CCA finds the linear projections $w_1^T X_1$ and $w_2^T X_2$ which have maximum correlation between them. Here, w_1 and w_2 are the canonical coefficients, $\sum_{11} \sum_{22}$ are the covariances of X_1 and X_2, and $\sum_{11}$ is the cross-covariance between the feature matrices.

4.3.2 Regularized Canonical Correlation Analysis (RCCA)

The basic version of the CCA does not work when the number of features (p_1, p_2) become larger than total number of samples (n). To handle this, regularization parameters $(\lambda_1$ and $\lambda_2)$ are added with the covariance matrices [50]. It is represented as **Equation 4.6** and **Equation 4.7**.

$$\sum_{11}' = \sum_{11} + \lambda_1 I_{p_1} \tag{4.6}$$

$$\sum_{22}' = \sum_{22} + \lambda_2 I_{p_2} \tag{4.7}$$

Here, λ_1 and λ_2 are the regularization parameters, I_{p_1} and I_{p_2} are identity matrices. The covariance matrices from **equation (4.5)** can be substituted with $\sum_{11}'$ and $\sum_{22}'$ for the regularized version.

4.3.3 Sparse Canonical Correlation Analysis (SCCA)

The interpretation of linear combinations often become impossible when datasets have large number of features. So, considering a sparse subset of the features is a feasible approach to handle it. In this case, the objective function needs to be maximized and it takes the following form of **Equation 4.8**.

$$corr(w_1^T X_1 , w_2^T X_2)$$
$$\text{where } ||w_1||^2 \leq 1, \|w_2\|^2 \leq 1, P_1(w_1) \leq c_1, \text{ and } P_2(w_2) \leq c_2 \tag{4.8}$$

Here, P_1 and P_2 are penalty function or sparse CCA criterion. The penalty functions provide sparse feature combinations and make the CCA deal with situations where the feature sets are large comparing to the number of samples. P_1 and P_2 can be lasso or fused lasso penalty functions. The parameters c_1 and c_2 are used to control the level of penalization.

4.4 Experiments and Results

4.4.1 Dataset Preparation

Gene similarity matrix and facial landmarks distance matrix are the two datasets. The number of rows/samples are the same in both the matrices. All the zero and constant valued features are removed from the matrices. The shape of gene similarity matrix and facial disease similarity matrix becomes 76×77 and 76×68 , respectively.

4.4.2 Hyperparameter Tuning

Some hyperparameters are tuned for each method. For the RCC analysis, the values of λ_l and λ_2 are needed to be found. In order to find the values, search is performed in a 5×5 matrix where λ_1 in one axis and λ_2 in another axis. The value of λ_1 and λ_2 are varied from 0.1 to 0.9. The value for the pair λ_1 and λ_2 is found to provide the best total correlation is 0.1 and 0.1.

So, this pair is selected to find the final canonical projections for both the train and test data. The R package: CCA is used for the RCC analysis.

For SCCA, the PMA package of R is used. The CCA method in the PMA package uses lasso penalty as parameter. The levels of parameterization are set using penaltyX and penalty_Y whose value should be in the range (0,1). To find the best values of penaltyX and penaltyY, search is performed in a 10×10 grid. The best correlation score is found when penaltyX = 0.8 and penaltyY = 0.8.

4.4.3 Total Correlation Scores

After tuning hyperparameters to the best values, the canonical correlation analyses (RCCA, SCCA) are performed, and total correlation scores are computed. The experiments are conducted under different number of output dimension (1 to 50) and the canonical coefficients ($w1$ and $w2$) are learnt for each of the output dimensions. This procedure is repeated for both the methods. After learning the coefficients, they are multiplied with the original dataset (train and test) to generate the projections. The total correlation scores are calculated from the projections of the test data to evaluate different methods using the linear_cca() method [51]. The results are illustrated in **Figure 4.3**.

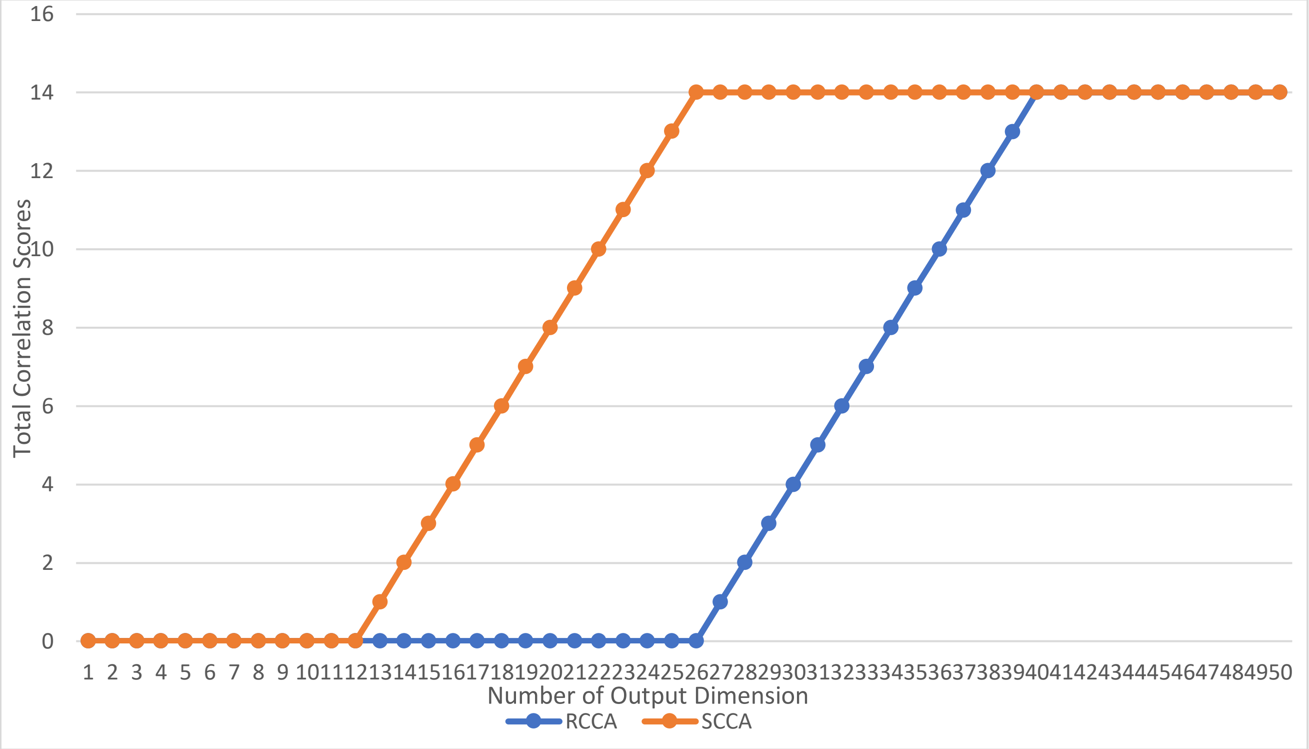


Figure 4. 2: Total correlation scores for RCCA and SCCA approaches. The x-axis represents the number of output dimensions, and the y-axis represents the corresponding total correlation scores.

From the above figure, it is visible that SCCA provides better correlation scores than RCCA. In both approaches, the correlation scores start from extremely small value, almost close to zero. But in case of SCCA, the value starts to rise sooner than RCCA. When the number of output dimension reaches to 25, the total correlation scores for SCCA starts to become constant. For RCCA, total correlation score starts to increase when the number of output dimension becomes 26. After some steps, the correlation scores become constant in RCCA. For SCCA method, the sparsity nature may correspond to the total correlation score.

4.5 Conclusion

The experiment in this chapter explores the association between the risk genes and facial feature of rare diseases. Total correlation score is calculated to compare the correlation among different CCA approaches. In future, some downstream analyses (classification, hierarchical clustering) can be performed on the datasets. Adding more data and increasing the size of the feature matrices might play a key role in such analyses. Since our data is limited, deep canonical correlation analysis is not performed assuming that enough data is not available. If more data are added later, deep canonical correlation analysis might be performed, and the intrinsic deep features may uncover more distinguishing feature leading to insightful correlation between genes and facial features.

Chapter 5: Deep Learning in Rare Disease Classification

5.1 Introduction

Many deep learning models such as Convolutional Neural Network (CNN), Deep Residual Network (DRN), Deep Autoencoder (DAE), Deep Belief Network (DBN) and Recurrent Neural Network (RNN) have been proposed over time and employed in visual recognition tasks. A typical CNN architecture consists of different combination of convolutional and pooling layers followed by one or more fully connected layers, presented in **Figure 5.1**.

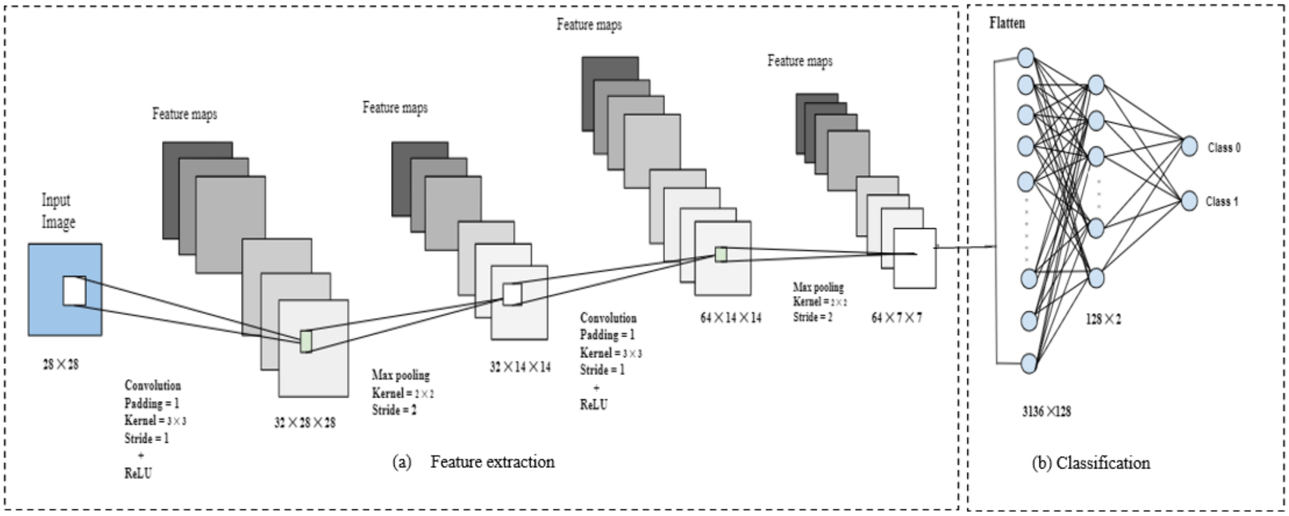


Figure 5.1: Architecture of a CNN. It consists of two parts. (a) Feature extraction: A 28×28 image is passed through convolution and pooling operations. Feature maps are generated after each operation. 64 feature maps are generated of size 7×7, at the end of feature extraction. (b) Classification: The $64 \times 7 \times 7$ feature map is flattened and a 3136×1 vector is generated. It is passed to a fully connected layer of 128 neurons. The 128 neurons are passed to fully connected layer of 2 neurons and probabilities are calculated for two classes. This way a 2-way classification takes place via CNN.

CNNs have shown exemplary performance in image classification, segmentation and retrieval related tasks [52]. The success behind CNN depends on its ability to identify the visual imaging patterns from the input pixels. Besides, the accessibility of large scale labeled training datasets has also accelerated the success.

The convolutional layer consists of a collection of convolutional kernels. The convolutional kernels divide the image into small blocks, also known as receptive fields. The kernels perform the convolution operation by multiplying the receptive fields with a particular set of weights [53].

$$f_l{}^k(p,q) = \sum_c \sum_{x,y} i_c(x,y).\, e_l{}^k(u,v) \tag{5.1}$$

Equation 5.1 shows a convolutional operation where, $i_c(x,y)$ is an element of the input image tensor I_c and $e^k{}_l\,(u,v)$ is the index of the k^{th} convolution kernel k_l of the l^{th} layer [54].

Pooling layers summarises the features present in the regions of feature maps generated by a convolutional layer and outputs the dominant response in this region. Pooling operation lessens the number of parameters to learn and reduces the feature map dimensions. The pooling operation is expressed as **Equation 5.2**:

$$Z_l{}^k = g_p(F_l{}^k) \tag{5.2}$$

where, $F_l{}^k$ is the input feature map, $g_p\,(.)$ denotes the pooling operation type and $Z_l{}^k$ is the pooled feature map. Max pooling is the most common operation among the others. Min pooling, average pooling, global average pooling etc. are some of the other pooling operations.

The activation function acts as a decision function in deep neural networks. It decides whether a neuron will be activated or not. Some commonly used activation functions are Sigmoid, Tanh, and ReLU.

The sigmoid activation function takes real numbers as input and its output is limited between 0 and 1. It is mathematically expressed as **Equation 5.3**:

$$f(x)_{sigm} = \frac{1}{1+ e^{-x}} \tag{5.3}$$

The Tanh function is same as sigmoid, except the output range. The output is restricted to between -1 and 1(**Equation 5.4**).

$$f(x)_{tanh} = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{5.4}$$

The ReLU converts the input value into positive number. It is mathematically represented in **Equation 5.5**.

$$f(x)_{ReLU} = \max{(0, x)} \tag{5.5}$$

Batch normalization standardizes the inputs to the layers for each mini-batch. The batch normalization layer standardizes the input layer distribution to address the internal covariance shift. Batch normalization for a transformed feature map F_l^k is presented in **Equation (5.6)**.

$$N_l^k = \frac{F_l^k - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \tag{5.6}$$

Where, N_l^k is the normalized feature map, μ_B and σ^2 represent the mean and variance of the feature map, respectively.

Even though there are several studies in the literature comparing different deep learning model performance [55], [56], [57], there is no well-explored comparative study on rare disease classification. In this chapter, this gap is addressed by providing a detail comparative study of deep learning models on rare disease classification. Several CNN models are applied to classify the facial photographs of children with rare disease with normal facial photographs of healthy children. As a contrast, the CNNs are employed to classify the facial photographs of children with rare disease with facial photographs of adults.

5.2 Materials and Methods

5.2.1 Datasets

We have developed a dataset to evaluate the models. This dataset consists only of facial photographs. The dataset is named as Dataset-A.

5.2.1.1 Dataset-A

The Dataset-A consists of 626 facial photographs of children. Among them, 480 facial photographs are the collected rare disease photos. The remaining 146 facial photographs belong to children who do not have rare diseases or any other diseases. In this thesis, such photographs will be mentioned as normal facial photographs. The 146 normal facial photographs in Dataset-A are collected from two resources: Dartmouth Database of Children's Faces (DDCF) [58] and FG-NET aging database [59]. The DDCF consists of facial photographs of 40 male and 40 female Caucasian children between the ages of 6 and 16 year. The models (children) are photographed from 5 different camera angles, posed with eight different facial expression: neutral, content (smile without teeth), happy (smile with teeth), sad, angry, surprised, and disgusted. The facial photographs of "neutral" male and female children aged between 6 and 12 years are considered as normal facial photographs for Dataset-A. Total 69 facial photographs (32 female and 37 male) are sampled from DDCF that satisfy the criterions of being neutral, age range of 6 to 12 years and frontal facing.

The FG-Net is an aging dataset consisting of 1002 images from 82 different subjects whose ages range between newborns to 69 years. The ages between 0 to 40 years are most populated in the database. Age 0 refers to newborn. The photographs of FG-Net are collected from personal collection, so the quality of the images depend on the photographer's skill, camera quality, photographic paper quality and condition of the photographs. For Dataset-A, only those photographs of the 82 subjects are chosen whose age fall within the range of 0 to 12 years. Total 77 images are sampled from FG-Net for Dataset-A that satisfy this criterion. Among the 77 facial photographs, 43 photos are newborns' (age 0). The rest of the photographs belong to several ages below the cut-off of twelve year: 7 photos of one-year-old, 12 photos of two-year-old, 6 photos of three-year-old, 5 photos of four-year-old, 2 photos of five-year-old and 2 photos of six- year-old.

The summary of the dataset is mentioned in **Figure 5.1**.

Table 5.1: Summary of Facial Photograph Dataset

Dataset	Total facial photographs	RD facial photographs	Healthy facial photographs
Dataset-A	626	480	146

5.2.3 Convolutional Neural Network Models

The widespread application of convolutional neural networks in image classification can be traced back to ImageNet Large Scale Visual Recognition Challenge (ILSVRC), an annual computer vision competition took place from 2010 to 2017, with the aim of developing improved computer vision methods. The challenge tasks (image classification, single object localization and object detection) were evaluated on a subset of publicly available dataset, ImageNet. This ImageNet subset consists of approximately 1 million images of 1,000 classes.

In the first five years of ILSVRC, massive success was achieved consistently by deep convolutional neural networks. Some of the milestone CNN architectures are, AlexNet (winner of 2012 ILSVRC with 15.4% error rate) [61], ZfNet (winner of 2013 ILSVRC, error rate: 11.2%) [62], VGGNet (winner of "classification+localization" category of ILSVRC 2014, error rate: 7.3%) [63], GoogleNet (winner of ILSVRC 2014, error rate: 6.7%) [64], ResNet (winner of ILSVRC 2015, error rate: 3.6%) [65]. Most of the models developed after 2015 were either improvement or ensemble over the previous ground-breaking models.

These CNN architectures provide general design principles which are adapted by machine learning practitioners to solve image recognition related tasks. These architectures act as rich feature extractor for image classification. In this chapter, the performance of some CNN architectures is compared in classifying rare disease and healthy facial photographs. Those architectures are chosen which has shown good performance in facial recognition tasks [29] [66].

The CNN architectures used in this study are AlexNet, VGG16, VGG19, ResNet-18, ResNet-34, ResNet-50, DenseNet-121 and MobileNet-V2.

5.2.3.1 AlexNet

AlexNet, proposed by Krizhevsky et al., consists of total eight layers: five convolutional layers, followed by three fully connected layers (**Figure 5.2**). AlexNet uses ReLU activation function to handle vanishing gradient problem. AlexNet has 60 million parameters, so overfitting is one major issue. It uses dropout in the fully connected layers to ignore some neurons randomly during training. AlexNet uses overlapping maxpooling in the convolutional layers and local response normalization (LRN) to improve generalization by reducing overfitting [61]. Besides, it performed data augmentation (translation, horizontal reflection) to scale-up the training data. In contrast to the previously proposed networks, AlexNet used large sized filters (11 × 11 and 5 × 5) in the initial layers. AlexNet has notable significance in the new era of innovative research in CNN applications.

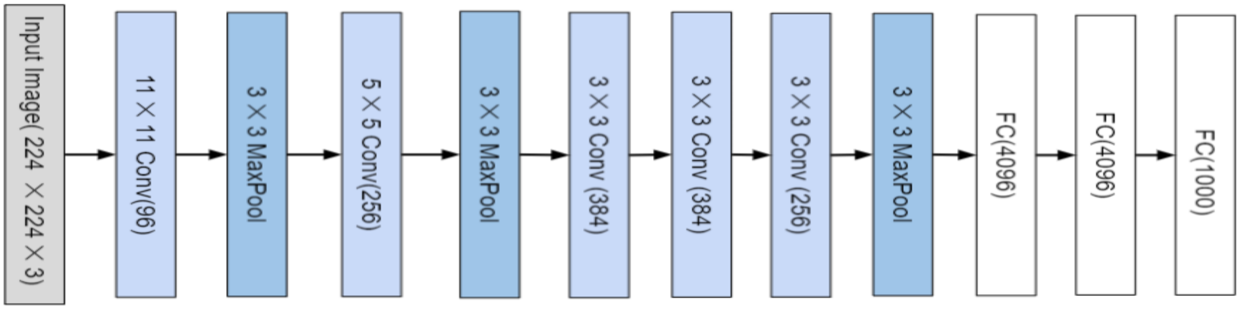


Figure 5.2: Architecture of the AlexNet

5.2.3.2 VGGNet

The success of AlexNet in the field of image recognition accelerated the research focusing on architectural design. Simonyan et al. developed an effective and simple design principle for

CNN, called VGGNet (**Figure 5.3**): VGG-11, VGG-13, VGG-16 and VGG-19. The numerical value here denotes the number of convolutional layers which are followed by three fully connected layers and softmax classification layer. The authors proved that the increasing number of layers can improve the final performance of networks to some degree. VGGNet did not employ the LRN layer used in AlexNet finding that its effect on CNN was not evident. It used small sized filters of 3×3 with stride 1, and maxpool layer of 2×2 with stride 2. It showed similar efficiency comparing to large sized filters ($11 \times 11, 7 \times 7$ or 5×5), used in earlier networks. This set a new research direction in working with small-size filters in CNN. One limitation of VGG is that it is computationally expensive due to ~138 million parameters.

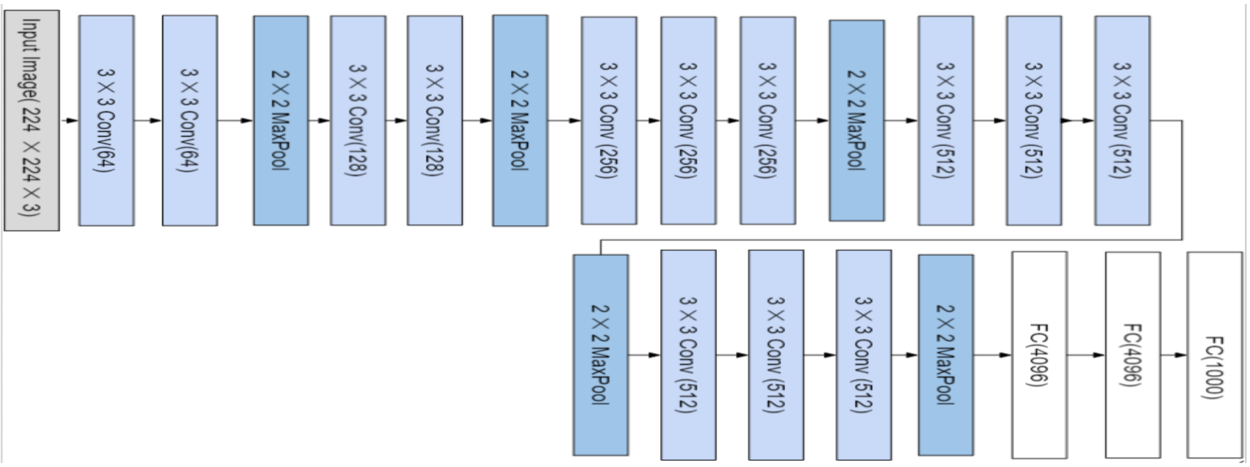


Figure 5.3: Architecture of VGG16

5.2.3.3 ResNet

It is expected that deeper networks show improvement over shallow ones since the deeper ones can extract more rich and complicated features than the shallow networks. However, the increase in depth makes the training task difficult and causes gradient vanishing, gradient exploding etc. The ResNet models, developed by He et al. alleviates the effect of vanishing gradient problem by utilizing identity connections or shortcuts around residual blocks to skip over some layers as illustrated in **Figure 5.4**.

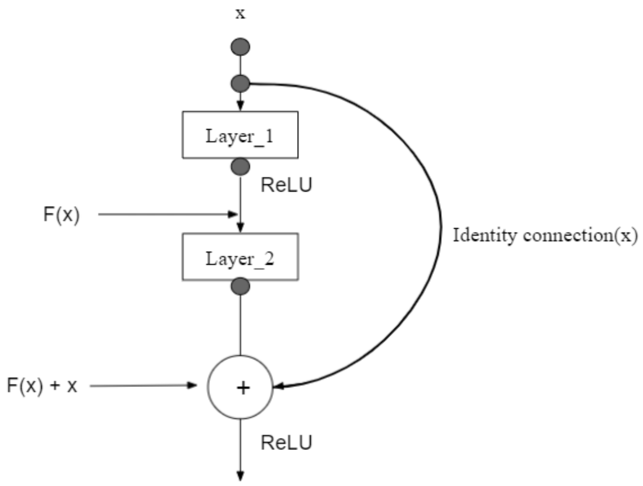


Figure 5.4: The illustration of ResNet identity connection. The input x is forwarded by the identity connection which is later added to the output of residual block $F(x)$. The final output is $F(x) + x$

The goal of identity connection is to allow gradient to flow through alternate shortcut way. Moreover, if any layer hampers the performance, it is skipped by regularization. ResNet uses two building blocks (identity block, convolutional block) to build the entire network.

Different version of ResNet exists such as ResNet-18, ResNet-34, ResNet-50, ResNet-101 etc. The ResNet-50 is the most common, consisting of 49 convolutional layers and one fully connected layer.

5.2.3.4 DenseNet-121

The Densely Connected Convolutional Network [67] (**Figure 5.5**), DenseNet is another step towards the increasing network depth. With the increasing number of layers, the path from input layer to output layer becomes very big which may cause the information to get vanished before reaching the last layer. DenseNet solves this problem by simplifying the connectivity pattern. In DenseNet, every layer is directly connected to every other layers. For N layers, DenseNet has $\frac{L(L+1)}{2}$ direct connections. So, the input of every layer in DenseNet is the concatenation of feature maps from previous layers. The layers of DenseNet are narrow, so lesser number of parameters (comparing to ResNet) are added. However, the feature map dimensions need to be same for concatenation. Inside each dense block, there is a transition layer consisting of a batch normalization layer, 1×1 convolution layer followed by 2×2 average pooling layer. The transition layer performs downsampling by applying the batch normalization.

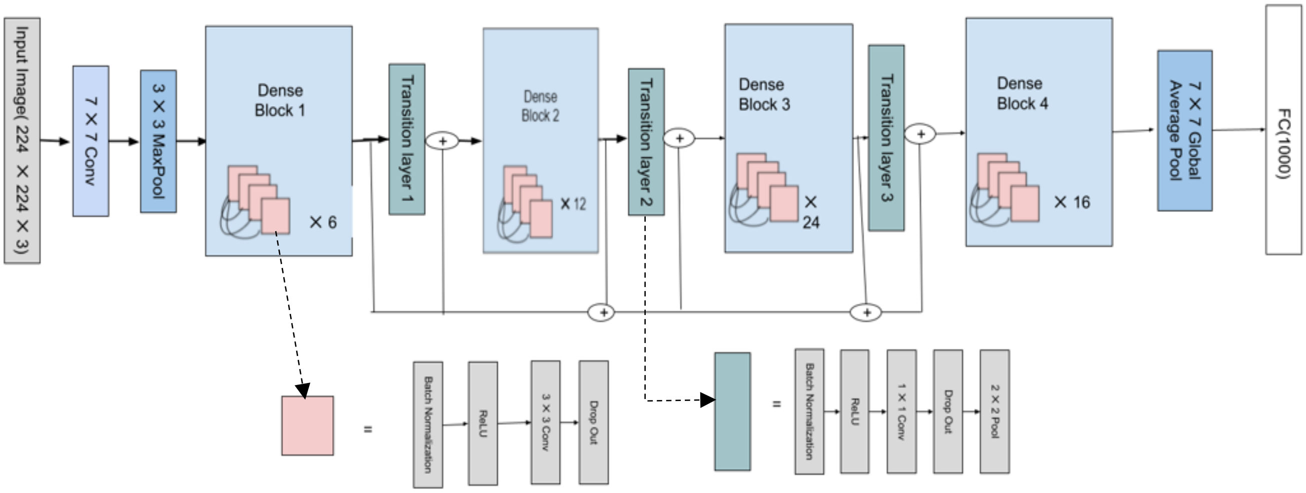


Figure 5.5: Architecture of the DenseNet121

5.2.3.5 MobileNet-V2

MobileNetV2 is a CNN architecture, developed by Google with the aim of performing well on mobile devices [68]. The architecture consists of two types of blocks: (1) Bottleneck residual block with stride of 1, and (2) Block with stride of 2 for downsizing. The residual block is presented in **Figure 5.6**. In each block, there are three layers. The first layer is a 1×1 convolution with a rectified linear unit (ReLU6), second layer is depthwise convolution and the third layer is also 1×1 convolution but without nonlinearity.

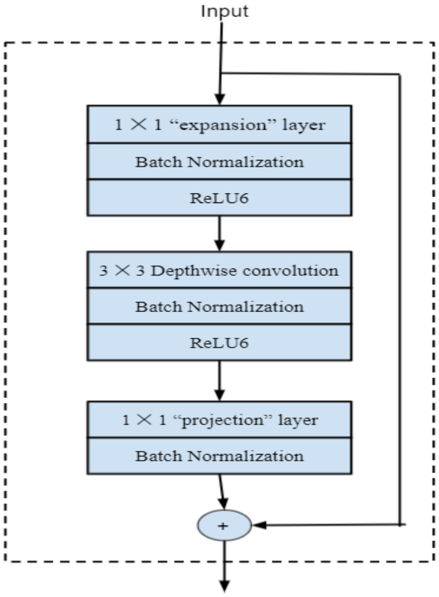


Figure 5.6: Bottleneck residual block

In bottleneck residual block, the first layer is a 1×1 convolution, it is called expansion layer. It expands the number of channel before it moves to the depthwise convolution layer. The expansion layer expands the input by the expansion factor. The default expansion factor is 6. The depthwise convolution layer filters the input, followed by a projection layer which projects the data in lower number of dimensions. The MobileNetV2 architecture consist of 19 bottleneck residual blocks, as depicted in **Figure 5.6.**

The summary of the CNN models is shown in **Table 5.2**.

Model	Publication	Main contribution	Parameters	Depth	Input Size
AlexNet	Krizhevsky et al. 2012	a. Utilizes Dropout, ReLU and overlap pooling b. GPUs NVIDIA GTX 580	60 M	8	$224 \times 224 \times 3$
VGG16	Simonyan et al. 2014	a. Homogeneous topology b. Uses small size kernels c.ReLU activation function is introduced after every convolutional layer	138 M	16	$224 \times 224 \times 3$
VGG19			143 M	19	
ResNet-18	He et al. 2016	a. Skip connection is introduced b. Residual learning is increased from hundreds to thousands of layers	11M	18	$224 \times 224 \times 3$
ResNet-34			21M	34	
ResNet-50			25M	50	
DenseNet-121	Huang et al. 2017	a. Blocks of layers are connected to each other b. Cross-layer information flow c.Reduces vanishing gradient problem	8M	121	$224 \times 224 \times 3$
MobileNet-V2	Sandler et al. 2018	a. Inverted residual structure b. Usual convolution layer is replaced with depth wise separable convolution.	4.25M	53	$224 \times 224 \times 3$

Table 5. 2: Architectures of the CNN models used in this thesis

5.2.4 Deep Transfer Learning (DTL)

Training a CNN from scratch for classification task requires a huge amount of data, which is not available. Therefore, the CNN architectures pre-trained on a large-scale dataset act as initializer for rare disease classifier training. When a neural network model is trained, the objective is to determine the correct weights of the network by multiple forward and backward iterations. By making use of the pre-trained models which are already trained on large datasets, the weights can be directly used, and the learning can be applied on another research problem of small dataset [69]. This approach is called transfer learning (TL). It is a powerful deep learning method in computer vision and the intuition is to transfer the knowledge/learning achieved from the source domain to a different but relevant problem domain, called the target domain. To sum up, transfer learning is based on the idea that the convolution base of CNNs extract reusable features and those features are transferrable to other tasks.

There are several ways transfer learning is applied. In this thesis, two commonly used transfer learning approaches are explored and employed.

1) **Fine tuning**: The weights of the CNN models are initialized with the weights from ImageNet, instead of random initialization. The final fully connected layers of the CNN models are removed and replaced with new classification layer according to the number of classes in the problem that is being tackled. At last, the CNN model is re-trained via backpropagation with the new training set.

2) **CNN as a fixed feature extractor**: The weights are initialized from the ImageNet and the final layers are replaced, like previous fine-tuning approach. But the convolutional base is kept frozen and only the final fully connected layers are re-trained. However, some convolutional blocks from

the rear can be unfrozen and re-trained with the fully connected layers. We froze the entire convolutional base in the experiments.

In this thesis, we have adopted the CNN architectures (discussed in Subsubsection **5.2.3**) pre-trained on ImageNet dataset. We expect that using the pre-trained CNN models through transfer learning approaches may improve the training speed and show better performance because the important image features are already learnt and transferred to the new task. Both transfer learning approaches are used in this experiment and the performance is analyzed. **Figure 5. 7** shows the schematic diagram of rare disease classification using transfer learning.

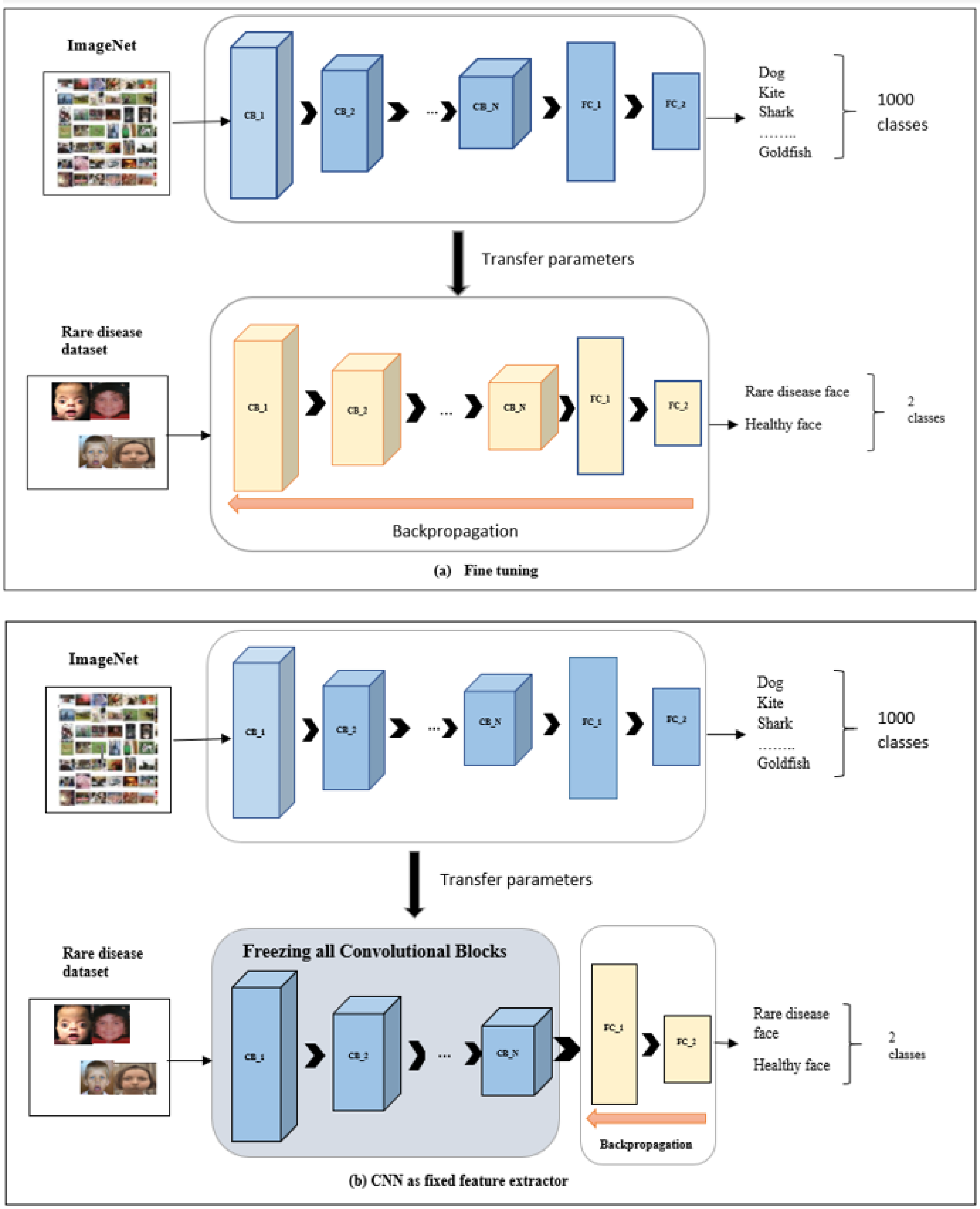


Figure 5. 7: Schematic diagram of rare disease classification using transfer learning. CB and FC represent convolutional block and fully connected layer, respectively. (a) Transfer learning approach by Fine tuning (b) Transfer learning approach treating CNN as a fixed feature extractor

5.2.5 Image Augmentation

This section presents the augmentation methods that are applied on the facial photographs while training the CNN. Image augmentation techniques expand the existing dataset by generating more images from the training samples. The objective is to handle the limitations of insufficient training data or the unbalanced data distribution in the dataset. Image data augmentation can be represented as **Equation 5.7**:

$$\emptyset : S \rightarrow \tau \qquad (5.7)$$

In this mapping, S is the original dataset and τ is the augmented set of S. The dataset is enlarged as the union of the original dataset and the augmented set as shown in **Equation 5.8**.

$$S' = S \cup \tau \qquad (5.8)$$

Common image augmentation techniques can be broadly categorized into geometric transformation and photometric transformation [70]. Geometric transformations transfer the pixel values of input image to new positions and thus modifies the geometry of an image. Such augmentation techniques include translation, rotation, reflection, zooming, mirroring, flipping, scaling etc. Photometric transformations alter the RGB channels of images to new values and such techniques include noise addition, filtering, color jittering, contrast adjustment etc.

In this study, rotation and noise addition are chosen as augmentation techniques to apply on the dataset. Along with increasing the size of the dataset, rotation makes the training images invariant to the changes in orientation and noise insertion helps the model to learn separating signal from noise in images. Both techniques are chosen considering the practical test case setting of this study. Because the facial images of test set might not always be perfectly frontal faced, sometimes the faces can be slightly leaned to any side and quality of the images may also vary. Gaussian, Speckle, Poisson, Localvar, Salt & Pepper etc. are some commonly used noises.

In this study, rotation is applied between 0° and 360° to the images during training. It changes the angles by which images appear in the training dataset. It increases variation in the training images and prevent the model from memorizing from training data. In this way, it helps to prevent possible overfitting.

The salt & pepper noise is added to the training images every time it is exposed to the model. So, the models become less capable of memorizing the training images which leads to increased robustness and lesser generalization error.

5.2.6 Traditional Machine Learning-based Methods

5.2.6.1 Support Vector Machine

Support Vector Machine (SVM) is a powerful method for classification. The goal of SVM is to construct a hyperplane in multidimensional space to separate the data points to their potential classes (**Figure 5.8**). The hyperplane needs to be positioned with the maximum distance to the data points. SVM generates optimal hyperplane in an iterative manner, which minimizes the error. The core idea behind SVM is to find a maximum marginal hyperplane that segregates the given dataset into classes in best possible way [20]. SVM uses kernel trick to transform an input data space into the required form.

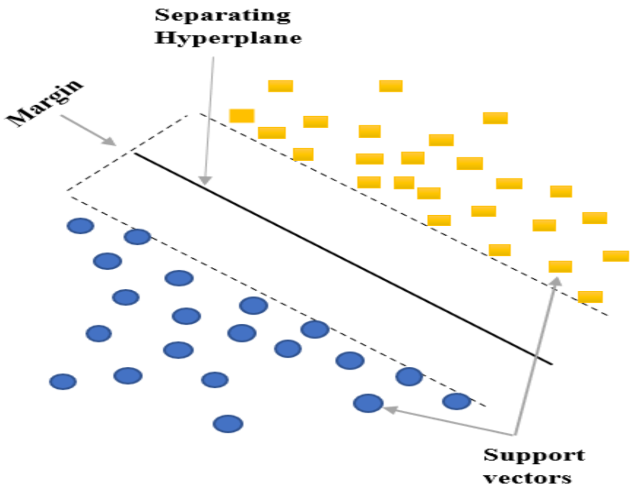


Figure 5.8: SVM classification algorithm. It separates the members of two classes using support vectors and separating hyperplane

The facial landmark feature data are applied in the SVM classifier. The landmark feature extraction and distance matrix generation process are elaborately explained from Subsubsection **4.2.1** to Subsubsection **4.2.3**. The landmark features extraction and slight modification to it for applying in SVM are precisely presented below:

1. From each facial image, 68 landmark points are calculated. One point is selected as a starting point and from that point, Euclidean distance is calculated to all the other points. As a result, 68×68 matrix is generated for each image. The distance matrix generation process is illustrated in **Figure 4. 1**. For each image, the 68×68 matrix is concatenated into a numpy array using the numpy package of python. The elements of the array are considered as features and given a label. For example, the arrays generated from rare disease facial photograph are given label 0 and others as 1.

2. The above step is repeated for all the images of the dataset. A SVM model with linear kernel is fit and classification is performed.

5.2.6.2 XGBoost Classifier

Extreme gradient boosting (XGBoost) is an advanced implementation of gradient boosting algorithm [71]. XGBoost is an ensemble learning algorithm which repeatedly generates new models and aggregates them into an ensemble model. The process begins by first computing residuals from the first base learner, building new models and adding residuals from the previous model to the ensemble of models.

XGBoost uses decision trees as base learners. The nodes of the trees can be referred to as some decisive questions with binary answers based on which the route is selected to reach the decisive point (leaf). XGBoost uses CART (Classification and Regression trees) which differs from the conventional decision trees. CARTs hold real-value score in their leaf nodes which indicates whether a particular sample belongs to a group. When a tree reaches the maximum depth (defined as a parameter, max_depth), the scores are converted into categories and decision is made.

The data applied on XGBoost for the experiment is same applied on the SVM. It is discussed in detail in the previous section. XGBoost is implemented using the scikit-learn Python libraries. This library provides fifteen parameters for XGBoost algorithm for tuning. Only four parameters (learning_rate, n_estimators, max_depth, subsample) is used considering that these might significantly affect the performance.

The learning_rate indicates how fast the model fits the residual errors. Its value is set to 0.01. The n_estimators specify the number of boosted trees, and it is fixed to 1000. Usually, the value is between few hundreds to thousands. The subsample refers to the portion of training set used for training each tree. Lower or higher value of subsample parameter may lead to overfitting or underfitting. Taking this into account, the value is set to 0.7. The maximum depth of the tree (max_depth) is fixed to 5.

5.2.7 Model Training and Evaluation

5.2.7.1 Hyperparameter tuning

The performance of eight CNN models (AlexNet, VGG-16, VGG-19, ResNet-18, ResNet-34, ResNet-50, DenseNet121 and MobileNetV2) and two machine learning models (SVM, XGBoost) are analyzed in this thesis. The transfer learning-based approaches are evaluated on all the CNN models. Next, data augmentation is performed on two models, DenseNet121 and ResNet50. All the models take $224 \times 224 \times 3$ sized input image. Several batch sizes (16, 25 and 32) are tested and the smallest batch size 16 is found as providing best result. The learning rate is set 0.0001 considering that a small learning rate will be able to capture the knowledge during fine tuning. During model training, the dataset is randomly split into training set (80%) and test set (20%). Next, the training set is again randomly split into 80% and 20%, where the 20% is considered as validation dataset. The images from the validation set are used to check the training performance after each epoch. So, the validation set is part of training. Cross entropy loss is used during training the model and Stochastic gradient descent (SGD) optimizer is used.

5.2.7.2 Cross-Validation strategy

The simplest model evaluation procedure is to divide a dataset into two parts and use one of them to train the model and use the remaining part to test the model. This is also called hold-out validation. But this procedure is effective when the dataset is large and representative of the problem formulation. The ideal choices for smaller datasets are k-fold cross validation or leave-one-out cross validation [72]. In this thesis, k-fold cross validation is applied.

In cross-validation, the training dataset is partitioned into k segments or folds of roughly equal size. The number of k is chosen as 5 in this thesis. The first fold is considered as the test dataset and the remaining $k - 1$ folds are used to train the model. Once the model is trained and evaluated on the test dataset, this process is repeated using the second fold and the other set of $k - 1$ folds. This process is repeated until each of the folds is used as test dataset. A total of k models are fit, evaluated and the performance of each fold is computed as the mean of the performance measures in each run. The dataset is shuffled every time before a fold is made, so that randomized folds are generated. Stratification is performed on the dataset so that the k folds are generated in stratified manner. The class distribution of the target variables from the dataset is determined and it is ensured that this distribution is preserved in the training and test dataset in each fold. This is one way to handle the imbalance that exists in our dataset.

Once the models are fit and evaluated k times, the average values are calculated from the performance measures generated in each fold.

5.2.7.3 Performance Measures

The most common and used evaluation metrics for image classification are accuracy, precision, recall, F1 score and ROC AUC score. These measures depend on the concepts of true positive, true negative, false positive, and false negative. Positive and negative indicate the classes (in this thesis, rare disease or healthy). True and false state whether the predicted class is same as the true class.

True Positive (TP): It refers to correct positive prediction. The case is positive, and it is also classified correctly.

False Positive (FP): It refers to incorrect positive prediction. This case is negative but falsely classified as positive.

True Negative (TN): It refers to correct negative prediction. The case is negative, and it is classified correctly.

False Negative (FN): It refers to incorrect negative prediction. The case is positive but falsely classified as negative.

Confusion matrix: A confusion matrix for a binary classification is a two-by-two matrix generated by counting the number of four outcomes of the classifier. It is shown in Error! Reference source not found..

Table 5. 3: Representation of a confusion matrix

		Actual class	
		Positive	Negative
Predicted class	Positive	TP (# of TPs)	FP (# of FPs)
	Negative	FN (# of FNs)	TN (# of TNs)

The rest of the evaluation metrics are computed from the confusion matrix.

Accuracy: It indicates the number of correct classifications divided by the total number of the dataset.

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

It refers to out of all the children, how many of them are correctly diagnosed as positive and how many as negative.

Precision (Positive Predictive Value): Precision refers to the total number of correct positive predictions divided by the total number of positive predictions. It is the ability of a classifier not to classify a class as positive which is negative.

$$Precision = \frac{TP}{TP + FP}$$

Recall (Sensitivity or True Positive Rate): It is calculated as the number of correct positive predictions divided by the total number of positives. It is a performance measure of the whole positive part of the dataset.

$$Sensitivity = \frac{TP}{TP + FN}$$

Specificity (True Negative Rate): It indicates the number of correct negative predictions divided by the total number of negative cases. It is the performance measure of the whole negative part of the dataset.

$$Specificity = \frac{TN}{TN + FP}$$

F1-Score: F1 score is the weighted average of precision and recall. F1-score is lower than accuracy because it embeds precision and recall into their computation.

$$F1 - Score = \frac{2 * (Precision * Recall)}{Precision+Recall}$$

AUC and ROC curve:

AUC-ROC curve is the performance measure for classification problems that represents how much the model is capable of distinguishing between classes. Receiver operating characteristic (ROC) is the probability curve and Area under the curve (AUC) represents the degree or measure of separability. The higher the AUC, the better the model is distinguishing between patients with diseases and no diseases.

5.3 Results and Discussion

This section presents the results of the CNN models and the traditional machine learning models in rare disease classification.

5.3.1 Performance on Transfer Learning as Fine Tuning

The eight pre-trained CNN models are evaluated on Dataset-A based on the fine tuning-based transfer learning approach, discussed in Subsubsection **5.2.4**, and the results are presented in **Table 5.4**. Since the models are pre-trained with ImageNet and fine-tuned, they already know how to detect some image features. Five-fold cross validation is performed during training with 100 epochs in each fold.

Table 5.4: Result of the pre-trained CNN models (Fine tuned) on Dataset-A with batch size 25

	Performance Measures					Parameters	
Model name	Accuracy	Precision	Recall	F1-score	AUC	Learning rate	Batch size
AlexNet	0.9397	0.94	0.94	0.94	0.57	0.0001	25
VGG-16	0.9031	0.91	0.90	0.89	0.95	0.0001	25
VGG-19	0.8714	0.89	0.87	0.85	0.91	0.0001	25
ResNet18	0.9095	0.92	0.91	0.90	0.93	0.0001	25
ResNet50	0.8857	0.90	0.88	0.87	0.95	0.0001	25
Densenet121	0.9159	0.92	0.91	0.91	0.96	0.0001	25
Mobilenetv2	0.9175	0.92	0.92	0.91	0.96	0.0001	25

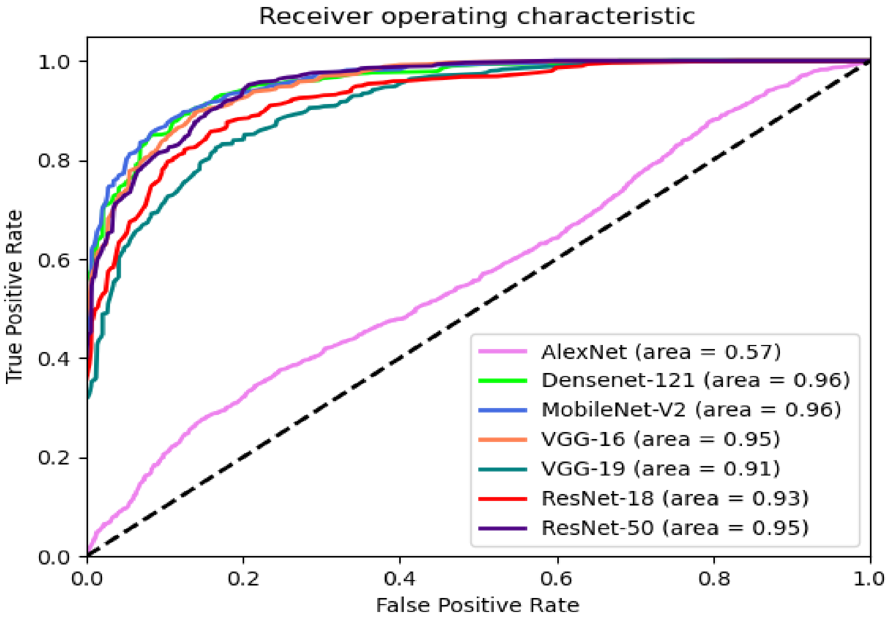


Figure 5. 9: ROC curves of the pre-trained CNN models on Dataset-A with batch size 25.

The receiver operating characteristic curve of the pre-trained CNN models is shown in **Figure 5. 9**. As finetuning is performed here, the weights of the models are initialized from ImageNet instead of random weight initialization. Except AlexNet, all the other models showed good classification performance as per AUC score.

The eight CNN models are analyzed in same experiment settings, with different batch size, 16 and 32. The performance of the models are presented in **Table 5.5** and the corresponding ROC curves are shown in **Figure 5.10** and **Figure 5.11.**

Table 5.5: Result of the pre-trained CNN models (Fine tuned) on Dataset A with additional batch sizes

	Performance Measures					Parameters	
Model name	Accuracy	Precision	Recall	F1-score	AUC	Learning rate	Batch size
AlexNet	0.9365	0.9401	0.9365	0.9353	0.97	0.0001	16
VGG16	0.8888	0.8929	0.8889	0.8781	0.93	0.0001	16
VGG19	0.9460	0.9465	0.9460	0.9444	0.99	0.0001	16
ResNet18	0.9238	0.9297	0.9238	0.9182	0.96	0.0001	16
ResNet34	0.9381	0.9418	0.9381	0.9344	0.94	0.0001	16
ResNet50	0.8841	0.8995	0.8841	0.8682	0.96	0.0001	16
DenseNet121	0.9159	0.9228	0.9159	0.9082	0.96	0.0001	16
MobileNetV2	0.9492	0.9498	0.9492	0.9475	0.99	0.0001	16
AlexNet	0.9222	0.9226	0.9222	0.9188	0.95	0.0001	32
VGG16	0.9048	0.9158	0.9048	0.8938	0.90	0.0001	32
VGG19	0.8762	0.8883	0.8762	0.8604	0.95	0.0001	32
ResNet18	0.8905	0.9011	0.8905	0.8763	0.91	0.0001	32
ResNet34	0.9095	0.9195	0.9095	0.8999	0.90	0.0001	32
ResNet50	0.9079	0.9183	0.9079	0.8981	0.94	0.0001	32
DenseNet121	0.8968	0.9075	0.8968	0.8853	0.94	0.0001	32
MobileNetV2	0.9016	0.9131	0.9016	0.8903	0.93	0.0001	32

The **Table 5.5** presents the results of the same set of experiments shown in **Table 5.4,** with different batch sizes. The models showed better performance with batch size 16 in almost all performance measures. Moreover, based on the AUC score, the performance of AlexNet significantly improved with both batch size 16 and 32, instead of batch size 25. **Figure 5.10** and **Figure 5.11** represent the ROC curves of the models with two different batch sizes.

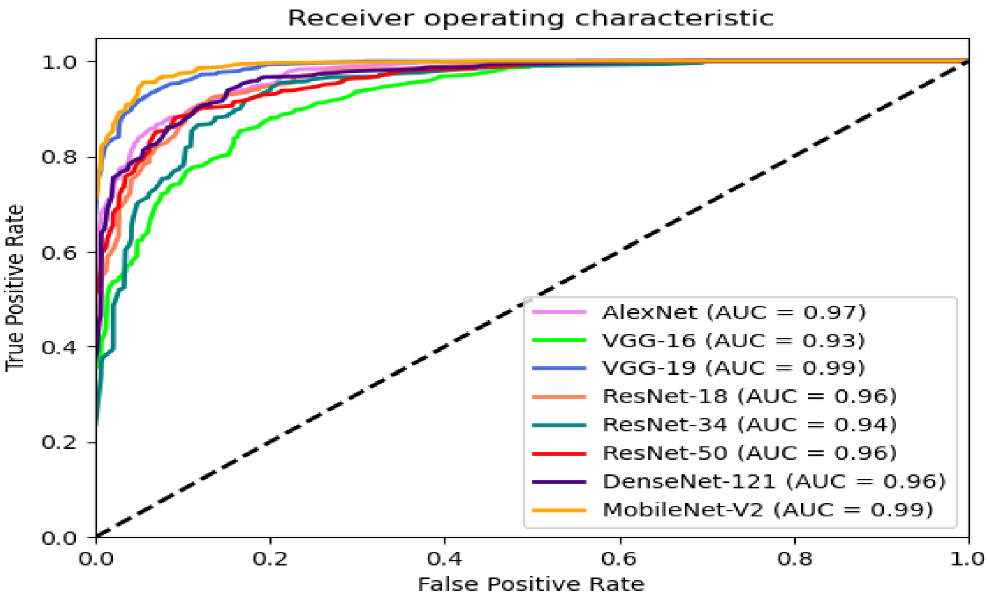


Figure 5.10: ROC curves of pre-trained CNN models on Dataset A with batch size 16. The models are pretrained on ImageNet and finetuned.

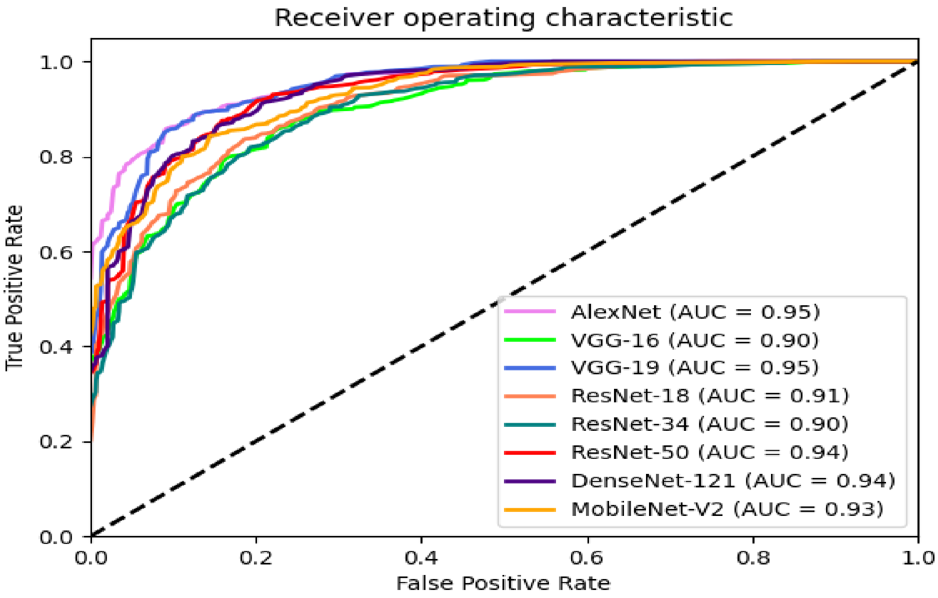


Figure 5.11: ROC curves of pre-trained CNN models on Dataset A with batch size 32. The models are pretrained on ImageNet and finetuned.

5.3.2 Performance on Transfer Learning as CNN as a fixed feature extractor

The performance of the eight CNN models is evaluated on Dataset-A using another transfer learning approach, where the convolution base of the CNN serves as a fixed feature extractor. The performance is evaluated on only batch size of 16, since it has shown better performance than other batch sizes as shown above. The results are presented in

Table 5.6.

Table 5.6: Results of the pre-trained CNN (as a fixed feature extractor) on Dataset-A

	Performance Measures					Parameters	
Model name	Accuracy	Precision	Recall	F1-score	AUC	Learning rate	Batch size
AlexNet	0.9063	0.9062	0.9063	0.9008	0.95	0.0001	16
VGG-16	0.9190	0.9261	0.9190	0.9123	0.90	0.0001	16
VGG-19	0.9063	0.9100	0.9063	0.8994	0.95	0.0001	16
ResNet18	0.8921	0.9059	0.8921	0.8778	0.91	0.0001	16
ResNet34	0.8921	0.9027	0.8921	0.8795	0.90	0.0001	16
ResNet50	0.9016	0.9132	0.9016	0.8901	0.94	0.0001	16
DenseNet121	0.8778	0.8953	0.8778	0.8578	0.94	0.0001	16
MobileNetV2	0.8809	0.8939	0.8809	0.8652	0.93	0.0001	16

Table 5.6 depicts the performance of CNN models when the convolution base is the fixed feature extractor. The convolutional blocks of the CNN models are kept frozen and only the classification layers are trained. Like previous section, five-fold cross validation is performed during training with 100 epochs in each fold.

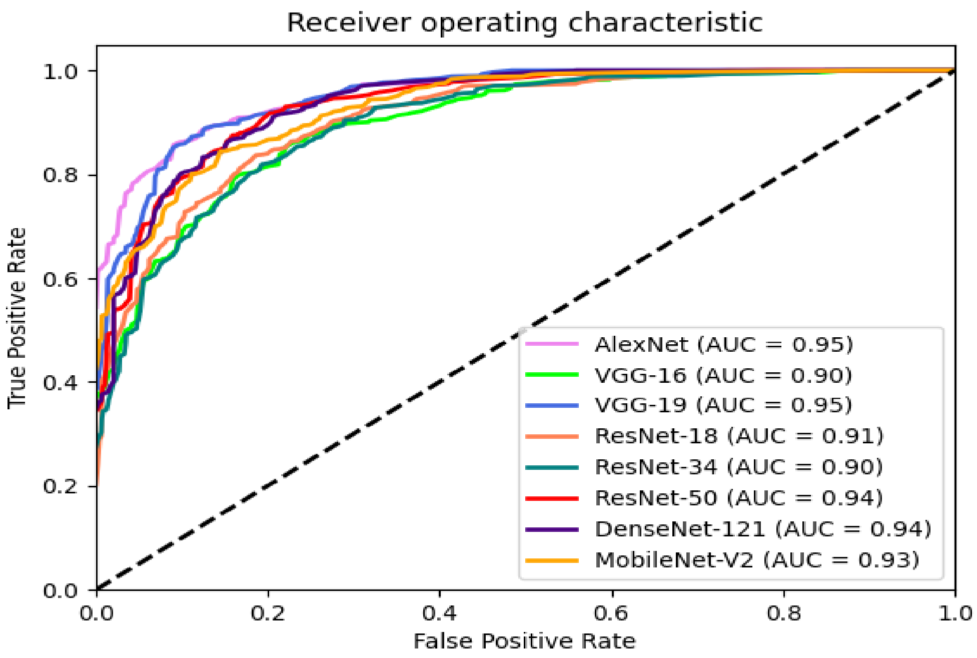


Figure 5.12: ROC curve of pre-trained CNN models (as a fixed feature extractor) on Dataset-A as a fixed feature extractor

It is visible from **Table 5.5** and **Table 5.6** that among the eight CNN models employing the fine-tuned transfer learning approach, six models achieved superior result in all performance measures than when the convolution base of the models were treated as fixed feature extractor. The VGG-16 achieved better result as a fixed feature extractor in four measures (Accuracy: 0.9190 vs. 0.8888, Precision: 0.9261 vs. 0.8929, Recall: 0.9190 vs. 0.8889, F1-score: 0.9123 vs. 0.8781) but not in terms of AUC (0.90 vs. 0.93). The ResNet50 also achieved better result as a fixed feature extractor in four measures (Accuracy: 0.9016 vs. 0.8841, Precision: 0.9132 vs. 0.8995, Recall: 0.9016 vs. 0.8841, F1-score: 0.8901 vs. 0.8682) except AUC (0.94 vs. 0.96)

The other seven models achieved better result in finetuning approach according to all performance measures. The MobileNetV2 model (fine tuned) showed best performance comparing to other models (Accuracy: 0.9492, Precision: 0.9498, Recall: 0.9492, F1-score: 0.9475, AUC: 0.99).

The superior performance of fine-tuning guided transfer learning-based models might lie to the fact that during fine tuning, the weights are initialized from ImageNet and the entire model is trained. On the other hand, in CNN as fixed feature extraction approach, the convolution blocks are frozen and only the classification layers are trained. Since the whole model was trained in fine-tuning approach, it learnt all the features and hence showed better performance.

5.3.4 Performance under Image Augmentation

Image augmentation is performed on the Dataset-A and its effect is observed over the performance of the two deep learning models, ResNet-50 and Densenet121. Two types of image augmentation (rotation and noise addition) are performed on the models, as discussed in the **Subsubsection 5.2.5**. Images on the training set are augmented in two scales. In first case, rare disease photographs are augmented to 5 whereas normal photos are augmented to 15. So, after augmentation the total number of images become 5,216. In another case, rare disease photographs are augmented to 10 and normal photographs are augmented to 30. After augmentation, the total number of images increases to 9,806. We have smaller number of normal facial photographs in Dataset-A comparing to rare disease photographs. For this reason, normal photos are scaled to double than rare disease photographs.

The augmentations are performed on the images of the training set, after the Dataset-A is split into training and test sets. This is how it is ensured that the test set does not get augmented. The performance of the is evaluated employing 5-fold cross validation. During each of the five folds, the dataset is split first, and augmentations are performed on the training set.

Table 5.7 and **Table 5.8** present the effects of augmentation on the models based on fine tuning and fixed feature extractor-based transfer learning approaches, respectively.

Table 5.7: Results of Image augmentation on pre-trained CNN (Fine-tuned) models on Dataset-A

	Performance Measures						Parameters	
Model name	Augmentation scale	Accuracy	Precision	Recall	F1-score	AUC	Learning rate	Batch size
ResNet-50	RD: 5 times, Healthy: 15 times	0.9571	0.9577	0.9571	0.9562	0.99	0.0001	16
ResNet-50	RD: 10 times, Healthy: 30 times	0.9556	0.9557	0.9555	0.9544	0.99	0.0001	16
DesneNet-121	RD: 5 times, Healthy: 15 times	0.9444	0.9491	0.9444	0.9446	0.95	0.0001	16
DenseNet-121	RD:10 times, Healthy: 30 times	0.9571	0.9581	0.9571	0.9565	0.97	0.0001	16

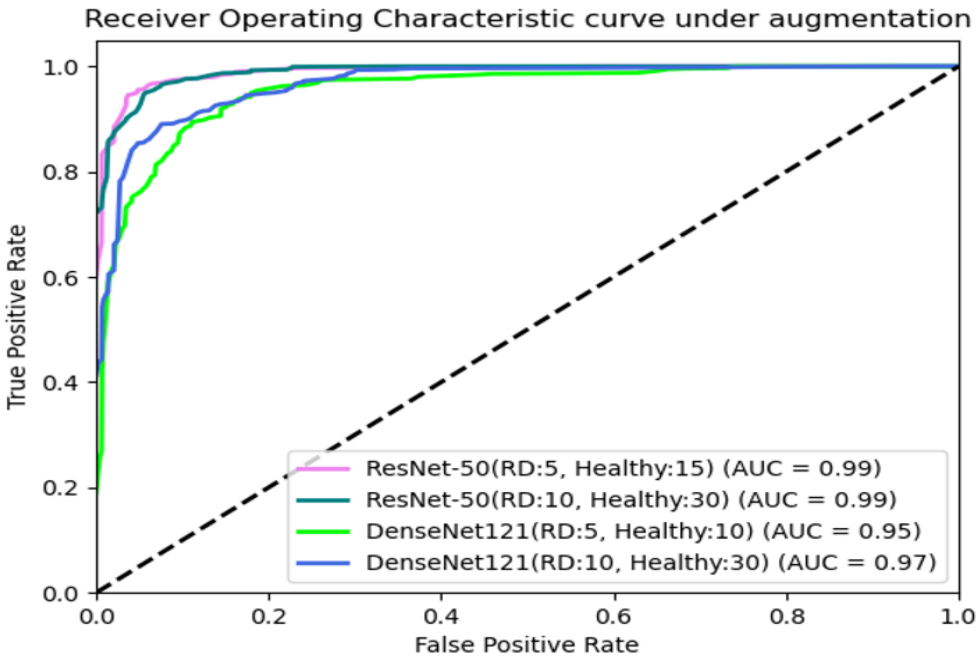


Figure 5.13: ROC curves of image augmentation on pre-trained CNN (Fine-tuned) models on Dataset-A

Table 5.8: Results of Image augmentation on pre-trained CNN (as a fixed feature extractor) models on Dataset-A

	Performance Measures						Parameters	
Model name	Augmentation scale	Accuracy	Precision	Recall	F1-score	AUC	Learning rate	Batch size
ResNet-50	RD: 5 times, Healthy: 15 times	0.9349	0.9355	0.9349	0.9348	0.98	0.0001	16
ResNet-50	RD: 10 times, Healthy: 30 times	0.9303	0.9313	0.9304	0.9304	0.98	0.0001	16
DesneNet-121	RD: 5 times, Healthy: 15 times	0.9387	0.9395	0.9387	0.9385	0.99	0.0001	16
DenseNet-121	RD:10 times, Healthy: 30 times	0.9332	0.9358	0.9332	0.9329	0.99	0.0001	16

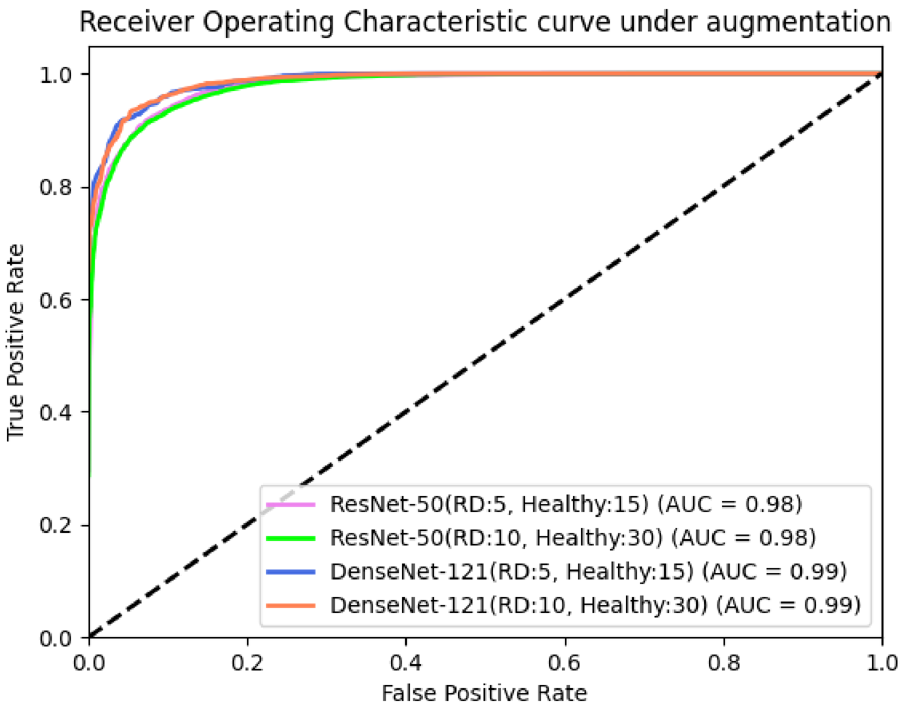


Figure 5.14: ROC curves of image augmentation on pre-trained CNN (as fixed feature extractor) models on Dataset-A

After comparing the results of CNN models without applying augmentation and after applying augmentation, it is visible that the CNN models achieved superior results on the augmented dataset. The performance of augmentation is evaluated on both the transfer learning approaches. In both cases, better result is observed on the augmented dataset in terms of all the measures.

The results from **Table 5.7 and Table 5.8** indicate that, even though the results are close, the fine-tuned CNN models showed overall better performance than the CNN models as fixed feature extractor. The fine-tuned ResNet-50 models (in both augmentation scales) outperformed the ResNet-50 models as fixed feature extractor. On the other hand, the fine-tuned DenseNet121 models (in both augmentation scales) achieved superior result than DenseNet-121(as fixed feature extractor) in terms of Accuracy (0.9571 vs. 0.9332), Precision (0.9581 vs. 0.9358), Recall (0.9571 vs. 0.9332) and F1-score (0.9565 vs. 0.9329), but achieved worse result in AUC (0.97 vs. 0.99)

5.3.5 Performance on Machine Learning Models

The machine learning based classification model results on Dataset-A are shown in **Table 5.9**. Like the previous deep learning-based models, five-fold cross validation is performed here as well.

Table 5.9: Result of Dataset A on SVM classifier and XGBoost classifier

Model	Accuracy	Precision	Recall	F1-score	AUC
XGBoost	0.834	0.822	0.842	0.822	0.88 ± 0.12
SVM	0.754	0.758	0.754	0.754	0.81 ± 0.07

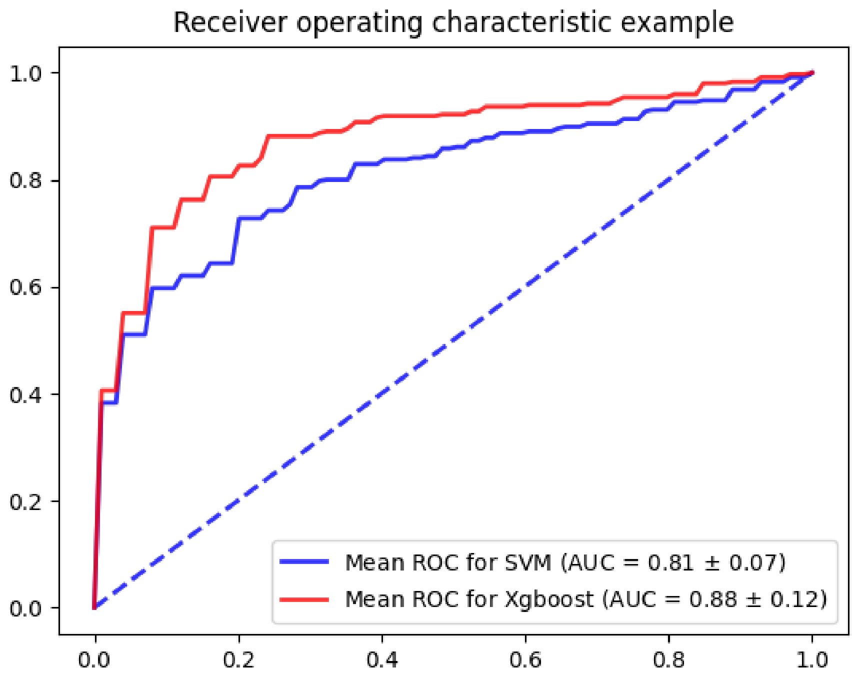


Figure 5.15: ROC curve of SVM and XGBoost on Dataset-A

It is visible from **Table 5.9** and **Figure 5.15** is that the performance of machine learning based classifiers is lower than deep learning based models. It is self-explanatory because the deep learning models extract high level features from images prior to classification. On the other hand, in machine learning methods, the features are hand-engineered. Only the distance matrices are given as input features in both SVM and XGBoost method. The results from **Table 5.9** indicates that XGBoost outperformed SVM in terms of all measures. XGBoost being a powerful model than SVM, the result is understandable. However, the hyperparameters of the XGBoost are considered by trial and error process. In future, if grid search is employed to set the optimal parameter, XGBoost might show more accurate result. On the other hand, the performance of the SVM model might improve if more features (statistical, holistic etc.) are extracted and fused with the geometric features.

5.4 Conclusion and Future Work

In this section, the performance of different CNN models is investigated in classifying rare disease affected children and healthy control. This work can be considered as a rare disease detection framework. Even though the fine tune transfer learning-based models outperformed other transfer learning approach, overall, the results from both the approaches were satisfactory. It indicates that transfer learning approaches are good selection in case of limited amount of data in facial disease identification. To some extent, it can handle the challenges of working with small datasets.

It is interesting to note that all the employed CNN models performed well in this classification task. The possible reasons might include choosing the suitable CNN models for recognition, using transfer learning approaches, performing augmentation and the problem being a binary classification. If it were a multiclass classification problem, the result might not be good.

In future, we will work on more data augmentation methods and gather more rare disease photographs. The system could be made more robust by engaging doctors or rare disease specialists in this research. The facial photographs from the datasets could be shown to them and their decision (whether the face has rare disease or not) will be considered as another performance measure. The proposed methods can be investigated further, and a mobile application can be developed. It could be easily used in any handheld devices and act as an effective tool for preliminary rare disease screening. The MobileNetV2 model is developed specifically focusing on portability. Besides, the fine-tuned MobileNetV2 model has shown best performance comparing to other models. So, this model can be extended further for this purpose.

Chapter 6: Conclusion, Limitations and Future Work

This study develops a new dataset of rare disease facial photographs and performs analysis on it through two aspects: (i) Correlation score calculation of the facial features and risk gene similarity estimation, and (ii) Classification of the rare disease from normal facial photographs.

In the first part of the thesis, 485 facial photographs of children with rare diseases are curated and stored in a database. In the next two parts of the thesis, the rare disease facial photographs are used to perform further analysis. The correlation between the facial features and risk gene semantic similarity is analyzed. In third part of the thesis, a systematic examination is performed on eight convolutional neural network architectures (transfer-learning based). The summary of this part is fundamentally a binary classification task which proves that a well-trained transfer learning guided model can detect rare disease given an input image of facial photographs. The results facilitate the body of knowledge in deep learning domain that transfer-learning based models, like many other applications can correctly classify rare disease when the input facial photograph dataset is quite small. Substantial experimental analysis is provided to support this decision.

The major challenge of this thesis is limited amount of data. There is no publicly available rare disease facial photograph database. In order to do the experiments and evaluation, we developed two datasets by ourselves combining the curated rare disease facial photographs and normal facial photographs from three different publicly available databases. The rare disease photographs are from publicly available sources whereas the normal facial photographs of children are collected from DDCF database where the images are taken as part of a research project and the FGNet database has images from personal collection. The quality of some images from the FGNet database is not so promising. It is hard to interpret how the differences between the facial images affected the model training and overall performance.

Another limitation is that multiclass classification is not performed here. The developed methods can only detect whether a facial photograph falls in the category of rare disease or healthy photos, but it is unable to report in which of the 104 rare disease categories the photograph belongs to. The reason behind the limitation is also the small number of curated facial photographs.

In future, we intend to collect more facial photographs of rare disease and normal facial photographs to conduct experiments in more realistic and neutral settings. The curated rare disease facial photographs can be considered as a benchmark dataset. Classifying the 104 rare diseases categories from limited amount of available data can be a promising research question for future.

References

[1] W. R. Evans and I. Rafi, “Rare diseases in general practice: recognising the zebras among the horses,” *Br. J. Gen. Pract.*, vol. 66, no. 652, pp. 550–551, Nov. 2016, doi: 10.3399/bjgp16X687625.

[2] B. Klimova, M. Storek, M. Valis, and K. Kuca, “Global View on Rare Diseases: A Mini Review,” *Curr. Med. Chem.*, vol. 24, no. 29, pp. 3153–3158, Sep. 2017, doi: 10.2174/0929867324666170511111803.

[3] R. Castro *et al.*, “Rare Diseases,” in *Handbook Integrated Care*, V. Amelung, V. Stein, E. Suter, N. Goodwin, E. Nolte, and R. Balicer, Eds. Cham: Springer International Publishing, 2021, pp. 763–782. doi: 10.1007/978-3-030-69262-9_44.

[4] A. Angelis, D. Tordrup, and P. Kanavos, “Socio-economic burden of rare diseases: A systematic review of cost of illness evidence,” *Health Policy Amst. Neth.*, vol. 119, no. 7, pp. 964–979, Jul. 2015, doi: 10.1016/j.healthpol.2014.12.016.

[5] *The Voice of 12,000 Patients. Experiences and Expectations of Rare Disease Patients on Diagnosis and Care in Europe*. EURORDIS - Rare Diseases Eu, 2009.

[6] P. U. Unschuld, *Huang Di Nei Jing Su Wen: Nature, Knowledge, Imagery in an Ancient Chinese Medical Text: With an Appendix: The Doctrine of the Five Periods and Six Qi in the Huang Di Nei Jing Su Wen*. University of California Press, 2003.

[7] B. Zhang, X. Wang, F. Karray, Z. Yang, and D. Zhang, “Computerized facial diagnosis using both color and texture features,” *Inf. Sci.*, vol. 221, pp. 49–59, Feb. 2013, doi: 10.1016/j.ins.2012.09.011.

[8] M. C. EL Rai, N. Werghi, H. Al Muhairi, and H. Alsafar, “Using facial images for the diagnosis of genetic syndromes: A survey,” in *2015 International Conference on Communications, Signal Processing, and their Applications (ICCSPA’15)*, Feb. 2015, pp. 1–6. doi: 10.1109/ICCSPA.2015.7081271.

[9] Y. Gurovich *et al.*, “DeepGestalt - Identifying Rare Genetic Syndromes Using Deep Learning,” *ArXiv180107637 Cs*, Jan. 2018, Accessed: Jun. 28, 2021. [Online]. Available: http://arxiv.org/abs/1801.07637

[10] H. S. Loos, D. Wieczorek, R. P. Würtz, C. von der Malsburg, and B. Horsthemke, “Computer-based recognition of dysmorphic faces,” *Eur. J. Hum. Genet.*, vol. 11, no. 8, pp. 555–560, Aug. 2003, doi: 10.1038/sj.ejhg.5200997.

[11] T. Vollmar *et al.*, “Impact of geometry and viewing angle on classification accuracy of 2D based analysis of dysmorphic faces,” *Eur. J. Med. Genet.*, vol. 51, no. 1, pp. 44–53, Jan. 2008, doi: 10.1016/j.ejmg.2007.10.002.

[12] J. Y. R. Cornejo and H. Pedrini, “Recognition of Genetic Disorders Based on Deep Features and Geometric Representation,” in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, Cham, 2019, pp. 665–672. doi: 10.1007/978-3-030-13469-3_77.

[13] Q. Ferry *et al.*, “Diagnostically relevant facial gestalt information from ordinary photos,” *eLife*, vol. 3, p. e02020, Jun. 2014, doi: 10.7554/eLife.02020.

[14] M. üge Çarıkçı and F. Özen, “A Face Recognition System Based on Eigenfaces Method,” *Procedia Technol.*, vol. 1, pp. 118–123, 2012, doi: 10.1016/j.protcy.2012.02.023.

[15] M. Anggo and La Arapu, “Face Recognition Using Fisherface Method,” *J. Phys. Conf. Ser.*, vol. 1028, p. 012119, Jun. 2018, doi: 10.1088/1742-6596/1028/1/012119.

[16] C. Janiesch, P. Zschech, and K. Heinrich, “Machine learning and deep learning,” *Electron. Mark.*, Apr. 2021, doi: 10.1007/s12525-021-00475-2.

[17] A. Paszke *et al.*, “Automatic differentiation in PyTorch,” Oct. 2017, Accessed: Jul. 21, 2021. [Online]. Available: https://openreview.net/forum?id=BJJsrmfCZ

[18] Y. Jia *et al.*, “Caffe: Convolutional Architecture for Fast Feature Embedding,” in *Proceedings of the 22nd ACM international conference on Multimedia*, New York, NY, USA, Nov. 2014, pp. 675–678. doi: 10.1145/2647868.2654889.

[19] J. Bergstra *et al.*, “Theano: A CPU and GPU Math Compiler in Python,” Austin, Texas, 2010, pp. 18–24. doi: 10.25080/Majora-92bf1922-003.

[20] “TensorFlow | Proceedings of the 12th USENIX conference on Operating Systems Design and Implementation.” https://dl.acm.org/doi/10.5555/3026877.3026899 (accessed Jul. 21, 2021).

[21] V. Gulshan *et al.*, “Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs,” *JAMA*, vol. 316, no. 22, p. 2402, Dec. 2016, doi: 10.1001/jama.2016.17216.

[22] A. Esteva *et al.*, “Dermatologist-level classification of skin cancer with deep neural networks,” *Nature*, vol. 542, no. 7639, pp. 115–118, Feb. 2017, doi: 10.1038/nature21056.

[23] J. T. Pantel *et al.*, “Efficiency of Computer-Aided Facial Phenotyping (DeepGestalt) in Individuals With and Without a Genetic Syndrome: Diagnostic Accuracy Study,” *J. Med. Internet Res.*, vol. 22, no. 10, p. e19263, Oct. 2020, doi: 10.2196/19263.

[24] Q. Zhao, K. Rosenbaum, R. Sze, D. Zand, M. Summar, and M. G. Linguraru, “Down syndrome detection from facial photographs using machine learning techniques,” in *Medical Imaging 2013: Computer-Aided Diagnosis*, Feb. 2013, vol. 8670, p. 867003. doi: 10.1117/12.2007267.

[25] P. Kruszka *et al.*, “22q11.2 deletion syndrome in diverse populations,” *Am. J. Med. Genet. A.*, vol. 173, no. 4, pp. 879–888, Apr. 2017, doi: 10.1002/ajmg.a.38199.

[26] T. Meng *et al.*, “Identifying Facial Features and Predicting Patients of Acromegaly Using Three-Dimensional Imaging Techniques and Machine Learning,” *Front. Endocrinol.*, vol. 0, 2020, doi: 10.3389/fendo.2020.00492.

[27] H. J. Schneider *et al.*, “A novel approach to the detection of acromegaly: accuracy of diagnosis by automatic face classification,” *J. Clin. Endocrinol. Metab.*, vol. 96, no. 7, pp. 2074–2080, Jul. 2011, doi: 10.1210/jc.2011-0237.

[28] X. Kong, S. Gong, L. Su, N. Howard, and Y. Kong, “Automatic Detection of Acromegaly From Facial Photographs Using Machine Learning Methods,” *EBioMedicine*, vol. 27, pp. 94–102, Jan. 2018, doi: 10.1016/j.ebiom.2017.12.015.

[29] B. Jin, L. Cruz, and N. Gonçalves, “Deep Facial Diagnosis: Deep Transfer Learning From Face Recognition to Facial Diagnosis,” *IEEE Access*, vol. 8, pp. 123649–123661, 2020, doi: 10.1109/ACCESS.2020.3005687.

[30] Y. Gurovich *et al.*, “Identifying facial phenotypes of genetic disorders using deep learning,” *Nat. Med.*, vol. 25, no. 1, pp. 60–64, Jan. 2019, doi: 10.1038/s41591-018-0279-0.

[31] P. Shukla, T. Gupta, A. Saini, P. Singh, and R. Balasubramanian, “A Deep Learning Frame-Work for Recognizing Developmental Disorders,” in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Mar. 2017, pp. 705–714. doi: 10.1109/WACV.2017.84.

[32] A. Singh and D. R. Kisku, “Detection of Rare Genetic Diseases using Facial 2D Images with Transfer Learning,” in *2018 8th International Symposium on Embedded Computing and System Design (ISED)*, Dec. 2018, pp. 26–30. doi: 10.1109/ISED.2018.8703997.

[33] D. Malm and Ø. Nilssen, “Alpha-mannosidosis,” *Orphanet J. Rare Dis.*, vol. 3, no. 1, p. 21, Jul. 2008, doi: 10.1186/1750-1172-3-21.

[34] “[From gene to disease; adrenocortical insufficiency, achalasia and disrupted tear secretion: Allgrove syndrome] - Abstract - Europe PMC.” https://europepmc.org/article/med/12497758 (accessed Jun. 18, 2021).

[35] V. Arora *et al.*, “Sialidosis type II: Expansion of phenotypic spectrum and identification of a common mutation in seven patients,” *Mol. Genet. Metab. Rep.*, vol. 22, p. 100561, Mar. 2020, doi: 10.1016/j.ymgmr.2019.100561.

[36] P. Resnik, “Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language,” *J. Artif. Intell. Res.*, vol. 11, pp. 95–130, Jul. 1999, doi: 10.1613/jair.514.

[37] J. J. Jiang and D. W. Conrath, “Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy,” *ArXivcmp-Lg9709008*, Sep. 1997, Accessed: Jun. 18, 2021. [Online]. Available: http://arxiv.org/abs/cmp-lg/9709008

[38] D. Lin, “An Information-Theoretic Definition of Similarity,” in *In Proceedings of the 15th International Conference on Machine Learning*, 1998, pp. 296–304.

[39] A. Schlicker, F. S. Domingues, J. Rahnenführer, and T. Lengauer, “A new measure for functional similarity of gene products based on Gene Ontology,” *BMC Bioinformatics*, vol. 7, no. 1, p. 302, Jun. 2006, doi: 10.1186/1471-2105-7-302.

[40] J. Z. Wang, Z. Du, R. Payattakool, P. S. Yu, and C.-F. Chen, “A new method to measure the semantic similarity of GO terms,” *Bioinformatics*, vol. 23, no. 10, pp. 1274–1281, May 2007, doi: 10.1093/bioinformatics/btm087.

[41] O. Spiga *et al.*, “Machine learning application for patient stratification and phenotype/genotype investigation in a rare disease,” *Brief. Bioinform.*, no. bbaa434, Feb. 2021, doi: 10.1093/bib/bbaa434.

[42] W. Ss, M. R, S. Jj, T. Me, and C. Mc, “[Orphanet: a European database for rare diseases].,” *Ned. Tijdschr. Geneeskd.*, vol. 152, no. 9, pp. 518–519, Mar. 2008.

[43] S. Köhler *et al.*, “Expansion of the Human Phenotype Ontology (HPO) knowledge base and resources,” *Nucleic Acids Res.*, vol. 47, no. D1, pp. D1018–D1027, Jan. 2019, doi: 10.1093/nar/gky1105.

[44] Gene Ontology Consortium, “The Gene Ontology: enhancements for 2011,” *Nucleic Acids Res.*, vol. 40, no. Database issue, pp. D559-564, Jan. 2012, doi: 10.1093/nar/gkr1028.

[45] L. du Plessis, N. Škunca, and C. Dessimoz, “The what, where, how and why of gene ontology—a primer for bioinformaticians,” *Brief. Bioinform.*, vol. 12, no. 6, pp. 723–735, Nov. 2011, doi: 10.1093/bib/bbr002.

[46] G. Yu, “Gene Ontology Semantic Similarity Analysis Using GOSemSim,” *Methods Mol. Biol. Clifton NJ*, vol. 2117, pp. 207–215, 2020, doi: 10.1007/978-1-0716-0301-7_11.

[47] D. E. King, “Dlib-ml: A Machine Learning Toolkit,” p. 4.

[48] O. Agbolade, A. Nazri, R. Yaakob, A. A. Ghani, and Y. K. Cheah, “Down Syndrome Face Recognition: A Review,” *Symmetry*, vol. 12, no. 7, Art. no. 7, Jul. 2020, doi: 10.3390/sym12071182.

[49] B. Thompson, *Canonical Correlation Analysis: Uses and Interpretation*. SAGE Publications, Inc, 1810.

[50] I. González, S. Déjean, P. G. P. Martin, and A. Baccini, “CCA: An R Package to Extend Canonical Correlation Analysis,” *J. Stat. Softw.*, vol. 23, no. 1, Art. no. 1, Jan. 2008, doi: 10.18637/jss.v023.i12.

[51] “VahidooX/DeepCCA: An implementation of Deep Canonical Correlation Analysis (DCCA or Deep CCA) with Keras.,” *GitHub*. https://github.com/VahidooX/DeepCCA (accessed Jul. 23, 2021).

[52] J. Schmidhuber, “Deep learning in neural networks: An overview,” *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015, doi: 10.1016/j.neunet.2014.09.003.

[53] J. Bouvrie, *1 Introduction Notes on Convolutional Neural Networks*.

[54] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, “A Survey of the Recent Architectures of Deep Convolutional Neural Networks,” *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5455–5516, Dec. 2020, doi: 10.1007/s10462-020-09825-6.

[55] Q. Liu, C. Feng, Z. Song, J. Louis, and J. Zhou, “Deep Learning Model Comparison for Vision-Based Classification of Full/Empty-Load Trucks in Earthmoving Operations,” *Appl. Sci.*, vol. 9, no. 22, Art. no. 22, Jan. 2019, doi: 10.3390/app9224871.

[56] A. Abubakar, M. Ajuji, and I. Usman Yahya, “Comparison of Deep Transfer Learning Techniques in Human Skin Burns Discrimination,” *Appl. Syst. Innov.*, vol. 3, no. 2, Art. no. 2, Jun. 2020, doi: 10.3390/asi3020020.

[57] M. M. Rahaman *et al.*, “Identification of COVID-19 samples from chest X-Ray images using deep learning: A comparison of transfer learning approaches,” *J. X-Ray Sci. Technol.*, vol. 28, no. 5, pp. 821–839, Jan. 2020, doi: 10.3233/XST-200715.

[58] K. A. Dalrymple, J. Gomez, and B. Duchaine, “The Dartmouth Database of Children’s Faces: Acquisition and Validation of a New Face Stimulus Set,” *PLOS ONE*, vol. 8, no. 11, p. e79131, Nov. 2013, doi: 10.1371/journal.pone.0079131.

[59] G. Panis and A. Lanitis, “An Overview of Research Activities in Facial Age Estimation Using the FG-NET Aging Database,” in *Computer Vision - ECCV 2014 Workshops*, vol. 8926, L.

Agapito, M. M. Bronstein, and C. Rother, Eds. Cham: Springer International Publishing, 2015, pp. 737–750. doi: 10.1007/978-3-319-16181-5_56.

[60] O. M. Parkhi, A. Vedaldi, and A. Zisserman, “Deep face recognition,” 2015, Accessed: Jun. 23, 2021. [Online]. Available: https://ora.ox.ac.uk/objects/uuid:a5f2e93f-2768-45bb-8508-74747f85cad1

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

[62] M. D. Zeiler and R. Fergus, “Visualizing and Understanding Convolutional Networks,” in *Computer Vision – ECCV 2014*, Cham, 2014, pp. 818–833. doi: 10.1007/978-3-319-10590-1_53.

[63] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” *ArXiv14091556 Cs*, Apr. 2015, Accessed: Jun. 24, 2021. [Online]. Available: http://arxiv.org/abs/1409.1556

[64] C. Szegedy *et al.*, “Going Deeper With Convolutions,” 2015, pp. 1–9. Accessed: Jul. 28, 2021. [Online]. Available: https://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Szegedy_Going_Deeper_With_2015_CVPR_paper.html

[65] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” 2016, pp. 770–778. Accessed: Jun. 24, 2021. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html

[66] T. Akter *et al.*, “Improved Transfer-Learning-Based Facial Recognition Framework to Detect Autistic Children at an Early Stage,” *Brain Sci.*, vol. 11, no. 6, p. 734, May 2021, doi: 10.3390/brainsci11060734.

[67] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely Connected Convolutional Networks,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, Jul. 2017, pp. 2261–2269. doi: 10.1109/CVPR.2017.243.

[68] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” 2018, pp. 4510–4520. Accessed: Jun. 24, 2021. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2018/html/Sandler_MobileNetV2_Inverted_Residuals_CVPR_2018_paper.html

[69] S. J. Pan and Q. Yang, “A Survey on Transfer Learning,” *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010, doi: 10.1109/TKDE.2009.191.

[70] I. Masi, A. T. Trần, T. Hassner, G. Sahin, and G. Medioni, “Face-Specific Data Augmentation for Unconstrained Face Recognition,” *Int. J. Comput. Vis.*, vol. 127, no. 6, pp. 642–667, Jun. 2019, doi: 10.1007/s11263-019-01178-0.

[71] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.

[72] S. Yadav and S. Shukla, “Analysis of k-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification,” in *2016 IEEE 6th International Conference on Advanced Computing (IACC)*, Feb. 2016, pp. 78–83. doi: 10.1109/IACC.2016.25.