A Deep Learning Approach for Automated Classification of Corneal Ulcers

by

Sumit Barua
19301070
Samit Saha
19301195
Jannatul Bulbuli
19301196
Akib Rahman
19301067
Md.Ibna Salam Fuad
17101539

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

Department of Computer Science and Engineering
School of Data and Sciences
Brac University
25th September 2023

Declaration

It is hereby declared that

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

Student's Full Name & Signature:

Samit Saha
19301195

Akib Rahman
19301067

Sumit Barua
19301070

Jannatul Bulbuli
19301196

Md.Ibna Salam Fuad
17101539

Approval

The thesis titled "A Deep Learning Approach for Automated Classification of Corneal Ulcers" submitted by

1. Sumit Barua (19301070)
2. Samit Saha (19301195)
3. Jannatul Bulbuli (19301196)
4. Akib Rahman (19301067)
5. Md.Ibna Salam Fuad (17101539)

Of Summer 2023 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science and Engineering on September 21, 2023.

Examining Committee:

Supervisor:
(Member)

Mr. Rafeed Rahman
Lecturer
Department of Computer Science and Engineering
Brac University

Co-Supervisor 1:
(Member)

Mr. Nabuat Zaman Nahim
Lecturer
Department of Computer Science and Engineering
Brac University

Co-Supervisor 2:
(Member)

Mr. Dibyo Fabian Dofadar
Lecturer
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam
Professor
Department of Computer Science and Engineering
Brac University

Chairperson:
(Member)

Dr. Sadia Hamid Kazi
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Ethics Statement

This thesis is dedicated to adhering to the utmost ethical principles while doing research. The values that we emphasize include respect for autonomy, beneficence, justice, privacy, integrity, and honesty. All study participants will be required to provide informed permission, and measures will be taken to ensure the protection of their privacy and confidentiality. The management of data will be conducted in accordance with rigorous protocols that prioritize both security and ethical considerations. All potential conflicts of interest are duly reported, and any ethical issues that may arise may be appropriately resolved by contacting the provided contact information. Our commitment is in the pursuit of research endeavors that have a beneficial impact on knowledge and society, while upholding principles of honesty and accountability.

Abstract

Eye Corneal Ulcer(ECU) has been demonstrated to be the second most common cause of treatable blindness worldwide, after cataracts. It is an extremely prevalent ophthalmic ailment and can cause severe visual impairment or perhaps total blindness.This thesis renders a comprehensive study on the automated classification of corneal ulcers using a deep learning approach. In this research, the SUSTech-SYSU dataset has been utilized which is obtained from Sun Yat-sen University's Zhongshan Ophthalmic Center, consisting of 712 images of patients with various types, grades and categories of corneal ulcers. These ocular surface images are captured after fluorescein staining, aiding as a valuable resource for the enhancement of deep learning models. The images in the dataset having dimensions of 2592 pixels in width and 1728 pixels in height, depicts close-up views of corneal abrasions under cobalt blue light during eye examinations, which is the particular type of image captured in this dataset. This thesis occupies the deep learning Convolutional Neural Networks (CNN) architecture, which includes InceptionV3, ResNet50 and VGG16 in order to create a pre-trained model for Eye-Corneal-Ulcer (ECU) image classification. In addition, a customized model is built to foster validation and test accuracy. The deep learning models are run on training and testing sets, enabling them to recognize unique criteria and patterns linked with different types of corneal ulcers. For data training, the dataset has been allocated by dividing it into separate folders based on the type of ECU images. Data augmentation is exerted by using the ImageDataGenerator to escalate the diversity of the dataset and improve model generalization. The dataset comprises 10,000 training images and 2,000 testing images for evaluating the model. In the customized model, a sequential architecture is implemented, including layers such as Conv2D, max pooling, batch normalization, flatten, and dense layers for feature extraction and classification. For multi-class classification, categorical cross-entropy is employed as the loss function, and the Adam optimizer is used. Hyperparameter tuning has been enacted using the validation set, encompassing various learning rates, batch sizes, and regularization techniques to optimize the performance of the model. The consequence of this research avails the development of automated corneal ulcer classification, potentially facilitating ophthalmologists in diagnosing and curing corneal infections more productively.

Keywords: Convolutional Neural Network; SUSTech-SYSU dataset; Eye corneal ulcer (ECU); Data Augmentation; ImageDataGenerator; Relu; Keras; Conv2D; max-pooling; Normalization; flatten; dense layers; Hyperparameter;

Dedication

We dedicated to the mentors who provided guidance and enlightenment, the family members whose steadfast support propelled my personal and academic endeavors, and the relentless search of knowledge that transcends limitations. Committed to the acquisition of knowledge and the perpetual quest for exploration.

Acknowledgement

Firstly, all glorification to the Almighty for whom our thesis has been accomplished without any major obstacle.
Secondly, to our Super-visor and co-super-visor teachers for their kind support and guidance in our work. They helped us whenever we asked for.
Lastly, the whole judging panel of BRACU. Our paper was accepted there, all the reviews and evaluation they gave helped us a lot in our later works.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The subject selected for our thesis is "Corneal Ulcer," a very distressing ocular illness known for its significant discomfort. Our research will focus on applying deep learning techniques to this area. Corneal ulcers have the potential to significantly impair visual acuity and may ultimately result in complete loss of eyesight. Corneal ulcers, also known as keratitis, may be defined as a pathological condition characterised by the presence of an open sore on the anterior surface, or cornea, of the eye. Ocular infections may manifest in the eyes, however some instances may not include the presence of bacterial, viral, or fungal pathogens. Instead, these infections may arise from conditions such as dry eye or neurotrophic corneas. Corneal ulcers often manifest as a reddened and painful ocular lesion. Visual acuity may be diminished and the presence of a yellow-white discharge might be seen, while a greenish discharge may potentially cause desiccation on the eyelids. The observed symptoms included heightened lacrimation, sensation of foreign body presence in the eye, photophobia, and in rare cases, edoema of the eyelid. There are three distinct classifications of corneal ulcers. One form of ulcers is fungal ulcers, which may be caused by the use of contact lenses. Insufficient cleaning of the lens cover may also result from a scrape or abrasion caused by contact with organic or plant matter. These occurrences are also often seen in individuals who use steroids, since the application of topical steroids may intensify the proliferation of fungal bacteria, leading to accelerated growth. The second category pertains to a bacterial ulcer. The development of this ulcer may be attributed to factors such as wearing contact lenses overnight, sleeping with them, or inadequate maintenance practises. Utilising an inappropriate solution, such as water instead of a multipurpose disinfectant, might potentially expose individuals to risks. Furthermore, it should be noted that lenses made with older technology exhibit reduced breathability, which has been linked to the development of ulcers. The third category includes viral ulcers, which may arise from the herpes simplex virus, known for causing cold sores, as well as the herpes zoster virus, responsible for shingles and chickenpox. A less prevalent kind of corneal ulceration, although very deleterious, is attributed to parasite infestation such as acanthamoeba. Acanthamoeba refers to a genus of tiny amoebae that mostly inhabit freshwater environments. These organisms provide particular challenges when it comes to contact lens use. This infection presents challenges in terms of both diagnosis and treatment. Neurotrophic ulcers, dry eye, and corneal hypoesthesia may contribute to the development of neurotrophic ulcers, characterized by the breakdown of the corneal tissue. Neurotrophic keratopathy in this case is characterized by its manifestation in the later stages. The observed ulcer is non-infectious in nature and exhibits distinct characteristics compared to conventional corneal ulcers.[5]

In this particular scenario, the implementation of an enhanced system that exhibits a higher level of accuracy in the spotting and classification of Eye Corneal Ulcers (ECU) via the usage of deep learning deftness and the Convolutional Neural Network (CNN) algorithm, accompanied by appropriate modifications, would be very advantageous. The Convolutional Neural Network (CNN), often known as ConvNet, is a specialized genre of Neural Network that is mostly applied in tasks relating to images, such as image ratification and pixel data processing. The convolutional layer that is included inside the system effectively reduces picture granularity while preserving essential information. The architectural configuration of a deep convolutional neural network is likely to be affected by well-established models that are pre-trained such as VGG16, ResNet50, and Inception V3. Apropos to gain optimal performance in the categorization of corneal ulcers, the model will undergo comprehensive tuning and optimisation using the provided data. Consequently, Convolutional Neural Networks (CNNs) are favoured for the contrivance of object identification and recognition.

1.1 Research Motivation

Corneal Ulcer disease is spreading worldwide nowadays. Corneal ulcers are ocular disorders of significant severity that may result in visual impairment and even blindness if not properly recognised and treated. The importance of timely and correct diagnosis cannot be overstated in terms of its impact on successful care and the prevention of serious consequences. At now, the identification of corneal ulcers mostly depends on the manual assessment conducted by ophthalmologists. However, this approach is associated with drawbacks such as time consumption, subjectivity, and potential variability influenced by the clinician's level of experience. [5]

In recent times researchers have been doing various types of surveys to identify the speardness factors, epidemic level, ages of affecting etc. to become aware of this disease. From many surveys, we got some valuable data which inspired us to research on this topic and reach on final solution. From 2007 to 2011, the Hospital University at Sains Malaysia did a survey on Corneal Ulcer-affected people Numbers. Below the survey data is mentioned.

Table 1.1: Table 1

Months	Ages	Male	Female
February	20-29	2	1
March	30-39	4	2
April	40-49	7	7
May	50-59	8	4
June	60-69	3	4
July	70-79	3	0
August	80-89	2	0
Total			47

This table it shows that the Corneal Ulcer (CU) disease affects the number of ages of males and females in an average year.[11]

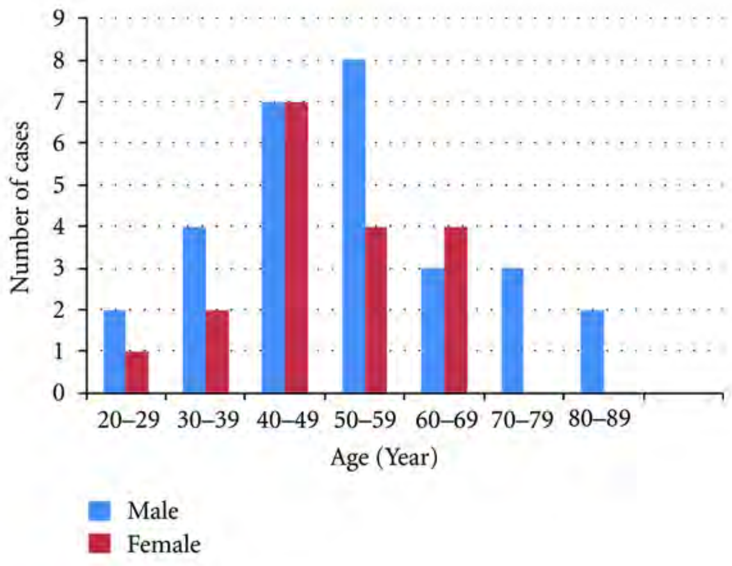


Figure 1.1: Survey Graph 1

Besides, In 2019, the University of Gondar Tertiary Eye Care and Training Center did a survey on eye corneal ulcer's number of cases.

Table 1.2: Table 2

Months	Number of Cases
February	2
March	2
April	6
May	5
June	4
July	4
August	1
September	4
October	2
Total	30

They show that the number of disease cases in every month of the year. This is the average prediction number for each year. [26]

Figure 1.2: Survey Graph 2

Additionally, there is also a another survey where they study on this disease. Their survey tells about from 2004 to 2009 infected corneal ulcer number, The graph is

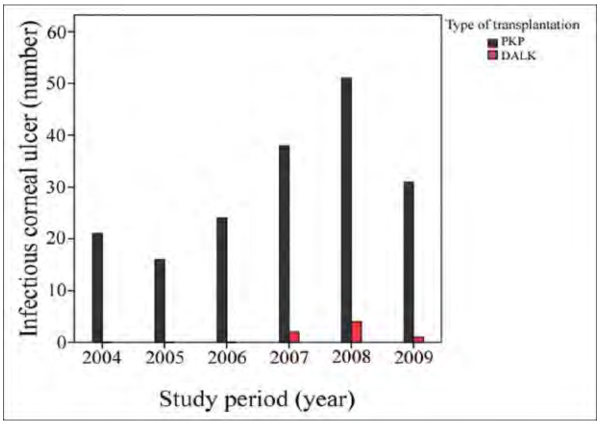


Figure 1.3: Survey Graph 3

Moreover, we have founded another survey which motivated us more to do research on this topic. This survey also tells about the number of diseases from December 2017 to January 2019.

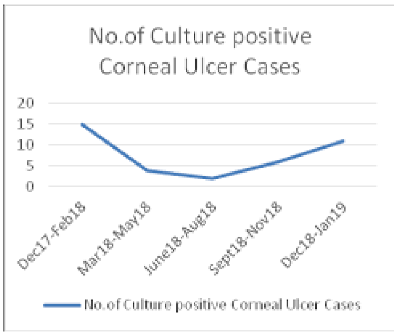


Figure 1.4: Survey Graph 4

Corneal ulcers are increasing around the world. However, people are still not aware of this disease. Most of the clinics still don't have the proper technologies to detect eye corneal ulcers accurately. They declare this disease a normal eye problem and give the wrong treatment. However, in some specific surveys, we can see that eye corneal ulcers are actually causing many people's eyes. [6]

1.2 Research Problem Statement

In the field of ophthalmology, the accurate and patient diagnosis of corneal ulcers is of paramount importance owing to effective patient care. However, the current diagnostic process often relies on subjective assessments by physicians, which can be time-consuming, subject to human error. and prone to inter-operator variability. Existing image segmentation algorithms, often relying on machine learning methods, require extensive training data to be effective. Therefore, the challenge lies in exacerbating a deep learning-based approach capable of motile classifying corneal ulcers into one of these three categories (pointlike, mixed point-flaky, or flaky) using available image datasets.

In recent times, the field of medical image resolution has seen the emergence of deep learning as a very effective technique, demonstrating notable achievements across a range of healthcare applications. Deep Learning (DL) is an advanced version of Machine Learning (ML) languages. DL models are used to classify image classifications which is also called computer vision. Corneal Ulcer datasets are mainly images. Through a multiclass of image we can detect the disease type, level, categories by Using deep learning models. Moreover, the usage of deep learning for the automated categorization of corneal ulcers remains significantly limited. The upliftment of an automated deep learning system aimed at categorizing corneal ulcers has the promise of transforming ophthalmic diagnostics via the provision of expedited, enhanced, and accurate outcomes. The objective of this perusal is to address the shortfall by employing deep learning deftnesses to improve the categorization of corneal ulcers. The development of a dependable and accurate automated model has the potential to enhance diagnostic efficiency for ophthalmologists, resulting in reduced time and resource requirements for human exams. Furthermore, the deep learning approach being presented aims to address the challenge of accurately distinguishing between several categories of corneal ulcers, including fungal, bacterial, viral, and maybe additional subtypes. The foremost quest of this erudition is to edify a very precise model for classifying ECU (Environmental Contaminant Units) into discrete groups, including fungal, bacterial, viral, and maybe additional types. [6]

1.3 Research Objective

The primary cornerstone of our perusal is to investigate the redaction stability of deep learning algorithms, specifically those utilizing Neural Networks, in the context of image classification across multiple classes. We look up to weigh the knack of these algorithms to maintain their accuracy when embedded to the training and testing both data, by passing the results through a sustainably presuming neural network. Furthermore, the convolutional neural network (CNN) image processing strait has exceptional redaction, achieving a near-perfect level of predicted accuracy. The motive of our lore was to ascertain the most appropriate algorithms for picture classification in order to get the highest level of accuracy in the findings. The objective of this project is to use the flair of deep learning in order to develop a robust, efficient, and dependable model for the automated categorization of corneal ulcers. Additionally, we enhance the whole patient experience, namely in the realm of optical diagnostics, while simultaneously facilitating significant advancements in medical image processing and ophthalmology. [12] Our objective also centered on comprehending which photographs exhibit higher recognition rates compared to others, in order to accurately determine the stage of the illness and effectively communicate this information to the patient. Our goal of this perusal is to debunk the effectiveness of deep learning techniques in automating the classification of corneal ulcers, therefore making a valuable benefaction to the expanding sector of medical picture probes. An effective automated system would not only provide assistance to physicians but also provide benefits to patients via the provision of early detection, prompt intervention, and perhaps reducing the risk of irreversible visual impairment. The act of categorizing facilitates the discernment of notable distinctions among the photographs within the collection. The dataset used in our study encompasses many classes, including different kinds, grades, categories, and phrases created from tweets, which were used for both training and testing purposes. Our objective is to use many modalities in the classification of photos and derive a comprehensive conclusion, so ensuring the legitimacy of the results. In order to accomplish our goal, we have followed the principles and research papers' guidelines that are on eye corneal ulcer detection using deep learning prior to us.

1.4 Research Contribution

The study conducted in the thesis titled "A Deep Learning Approach for Automated Classification of Corneal Ulcers" is a substantial contribution to the discipline of automated disease spotting and medical picture exposition. The primary intent of this perusal is to use deep learning methods for the categorization of corneal ulcers. The research demonstrates an improvement in precision achieved via customization by comparing the efficacy of pre-trained models with that of models designed specifically for the task at hand.

This research contributes to the progress of artificial intelligence in the sector of ophthalmology by the infliction of deep learning methods to the particular task of categorizing corneal ulcers. The use of deep learning deftness has yielded favourable results across a range of medical imaging tasks, and the findings of this work provide further evidence of its effectiveness in the detection and diagnosis of ocular illnesses. The results and methodology of the study may serve as a basis for further investigation into the use of deep learning in the analysis of medical-related images and the treatment of different ocular diseases. An additional noteworthy contribution is the headway of a tailored deep-learning model specifically designed for the purpose of categorizing corneal ulcers. Although pre-trained models have shown to be valuable in many picture identification tasks, this work underscores the need for fine-tuning algorithms to cater to the distinct features and challenges associated with corneal ulcer images. The superior performance shown by the custom-built model underscores the significance of domain-specific knowledge and optimisation, resulting in enhanced precision and consistency in illness categorisation. [24]

The research's demonstration of enhanced corneal ulcer categorization accuracy is its most notable contribution. The custom-built model performed better than the pre-trained models, achieving a validation exactitude of 90% which is sublime than the other pre-trained ones. This notable increase in accuracy suggests that developing deep learning models specifically for the challenge of categorizing corneal ulcers can result in a more precise and reliable diagnosis. The increased accuracy is anticipated to have real-world applications in the medical field, where trustworthy automated diagnostics can help medical practitioners treat patients with corneal ulcers promptly and accurately. As the automated classification of corneal ulcers may simplify the diagnostic process and possibly reduce human error in disease diagnosis, the research's findings have significant clinical implications. [31] The custom-built model offers itself as a valuable tool to aid ophthalmologists and other medical personnel in their decision-making process, enabling quicker and more accurate diagnoses.

The study also advocates for the adoption of deep learning-based methodologies in clinical applications, which may improve treatment for patients and the results of diagnosis.

In summary, this thesis significantly contributes to the domains of medical image deciphering and ophthalmology. The study elucidates the probable benefits of integrating artificial intelligence (AI) technology into healthcare environments and provides valuable insights for improving automated illness diagnosis via the exertion of a personalized deep-learning model, which exhibits superior classification accuracy.

Chapter 2

Literature Review

2.1 Related Works

In [6] this paper, the primary aim of the research conducted by Chen, Pires, Tseng, and colleagues (2019) is to assess the potency of amniotic membrane transplantation as a potential alternative treatment for neurotrophic corneal ulcers. In this perusal, the application of Amniotic Membrane Transplantation (AMT) has demonstrated successful outcomes in a cohort of 15 patients suffering from neurotrophic corneal ulcers and impaired visual acuity. The utilization of AMT has proven effective in addressing chronic corneal epithelial defects and ulcers resulting out of various conditions such as keratoplasty, herpes zoster ophthalmicus, diabetes, radiation, and acoustic neuroma excision, thereby inducing a neurotrophic state. The research encompassed a sample of 16 ocular cases that had undergone several surgical interventions, with a majority of 68.7% having received multiple intraocular procedures. The main objective was to reinstate the anatomical integrity; nonetheless, there was a high incidence of graft failure. Vision was at risk due to pre-existing ocular disorders. Every individual's visual acuity was found to be below 20/200. In spite of the administered therapies, it was observed that all ocular specimens continued to exhibit persistent epithelial defects and stromal ulcers. Following the transplanting of the amniotic membrane, there was a notable enhancement in visual acuity in eight eyes, accounting for 50% of the cases. Conversely, in six eyes, representing 37.5% of the cases, visual acuity remained unaltered.

The objective of the study conducted by M. Srinivasan, C. Gonzales, C. George, V. Cevallos, J. Mascarenhas, B. Asokan, J. Wilkins, G. Smolin, J. Whitcher is to determine the specific etiological agents responsible for infection and to investigate the epidemiological properties and risk facets associated with corneal ulceration. The research investigated a cohort of individuals diagnosed with central corneal ulcers by the utilization of a biomicroscope. Subsequently, the ulcers were cultured on different types of media for further analysis. The bacterial cultures underwent incubation, assessment, and subsequent disposal in the event of absence of growth. The identification of fungi was conducted by assessing their colony features and examining their appearance using lactophenol cotton blue staining. The study collected sociodemographic data and information related to risk factors. Additionally, corneal cultures and scrapings were conducted. During a duration of three months, a total of 434 individuals presenting with central corneal ulcers were assessed. A total of

65.4% of the participants exhibited a documented history of corneal damage. The results of the cornea culture analysis indicated a good outcome in 68.4% of the patients. Among the cases examined, a significant proportion of 47.1% presented with exclusive bacterial infections, while 46.8% exhibited exclusive fungal infections. A smaller subset of 5.1% displayed mixed infections involving both bacteria and fungi, whilst a little 1.0% demonstrated pure Acanthamoeba cultures. Streptococcus pneumoniae emerged as the predominant bacterial pathogen. [5]

In the paper [24], this research examines the global prevalence of reversible blindness, specifically focusing on the second most prevalent cause following cataracts. Additionally, it explores the dormant of deep learning algorithms in discerning the most diagnostically significant characteristics of a disease based on visual representations. The present work used an algorithm, originally devised by Williams and Long, to investigate the ocular sub basal nerve plexus in individuals diagnosed with diabetic neuropathy. However, it is vital to allude that the system's training process solely relied upon photographs of the pupil region. In this study, the comprehensive ocular surface picture is utilized to detect and diagnose peripheral corneal and limbal diseases. The research was granted approval by the institutional review board (IRB) of the Shanghai Eye, Ear, Nose, and Throat Hospital, with the reference number EENTIRB20170607. A framework for domain taxonomy was devised in order to systematically categorize diseases into manageable groups, organized hierarchically inside a pie structure. A total of 5325 high-resolution images of the ocular surfaces were collected for the purpose of training and validation. A corneal sickness encompasses various conditions that impact the cornea, including non-infectious keratitis and degenerative disorders. The proposed deep learning system comprises various classifiers capable of acquiring the ability to handle multiple tasks and labels that encompass multiple levels of classification for eye diseases. The deep learning algorithm, as suggested, exhibits superior performance compared to a group of ten ophthalmologists in the diagnosis of a corneal inflammatory condition. This is achieved by the use of ocular surface pictures, specifically those depicting infectious and non-infectious keratitis. In terms of the region under the recipient operating characteristic curve (AUC), this model demonstrates superior performance compared to Resnet34, Inception-v3, DenseNet, and ensemble across a wide range of corneal disorders. The research utilized an algorithmic approach to detect corneal abnormalities in the outpatient clinics of two tertiary eye hospitals. This algorithm was able to determine the presence of the disease by using previously unobserved visual data obtained from each diagnostic test. The algorithm demonstrated a notable level of diagnostic accuracy when implemented in outpatient clinics for the purpose of detecting corneal diseases. Its primary objective was to distinguish between commonly misdiagnosed illnesses, particularly those that impact the center cornea. Additional research is necessary to ascertain whether the algorithm has the potential to improve patient care and outcomes.

From [12], Bacterial keratitis has a prevalence rate of 0.3% among patients in the United States, with the primary treatment option being the administration of fourth generation fluoroquinolones. Corneal ulcers can arise due to various factors, such as recent corneal surgery, trauma, contact lens usage, ocular surface sickness, and prolonged utilization of topical pharmaceuticals. In the year 2010, a total of 736 riddens received a prognosis of keratitis, which led to the initiation of a microbiological investigation. More than 55% of the patients had already initiated the use of topical antibacterial treatment. Risk factors for the condition under consideration encompass a history of trauma, the utilization of contact lenses, and the recurrent application of topical steroids. In cases where there is significant thinning or perforation of the cornea, microbiological examinations are conducted, followed by an attempt at corneal gluing. Keratoplasty surgery is indicated as a treatment option in cases when there is a suspicion of a parasite infection requiring repair. Fluoroquinolones are commonly prescribed for their efficacy in providing broad-spectrum coverage against Gram-negative bacteria, whereas boosted antibiotics are often preferred for their ability to effectively target Gram-positive bacteria. Agar-agar plates are exclusively offered in cases when there is a suspicion of a parasite infection. [16] The CCCCL has not been authorized by the US Food and Drug Administration, hence it is not yet obtainable at the clinic. Potential advancements in molecular diagnostics, such as the utilization of multiplex PCR and imaging techniques, have the potential to enhance the identification and treatment of microbial keratitis.

According to [31], corneal ulcers are very dangerous diseases that come from a variety of infections caused by parasites, viruses, or bacteria. Early identification can decrease the likelihood of encountering the visually impaired. In this paper, there are two methods being proposed to segment corneal ulcers, techniques of image processing and semantic segmentation with deep learning. SUSTech-SYSU which is a publicly available dataset is being used in this paper. There are 354 photos with labels indicating the location of the corneal ulcer area. On the basis of the test data, the pre-trained CNN models were used for training and testing. 2016's ImageNet champion was the Resnet-18 model, which is the type of ResNet used in this study. The data were split into 70%: and 30% (training: testing) for the pre-trained ResNet18. The images were resized to $224 \times 224 \times 3$.

In [19] Neural networks have undergone significant advancements, leading to the development of deep learning algorithms. Prominent examples of such models include GoogLeNet8, ResNet 9, and VGGNet 10. Inception modules are employed to capture local characteristics, while ResNet addresses challenges associated with deep neural networks, such as the problem of disappearing gradients. There is a requirement for a methodology to assess ulcerative keratitis in canines that effectively encompasses environmental variables and their impact on clinical and epidemiological characteristics. A research investigation was carried out utilizing the corneas of 281 canines, from which a total of 368 corneal images were gathered over the period spanning from 2015 to 2017. The models successfully differentiated between superficial and deep corneal ulcers, with ResNet and VGGNet having an accuracy rate of 90%. The performance of GoogleNet was comparatively weaker. In this study, a Convolutional Neural Network (CNN), a recently developed deep-learning algorithm for picture classification, was employed to discern the severity of corneal ulcers in canines. The models were trained using images, and the work was executed using TensorFlow, an open-source software tool developed by Google. In the classification of shallow and deep corneal ulcers, it was shown that three CNN models had a significant performance advantage over one another, with an accuracy improvement above 90%. One notable constraint of the study pertains to the reliance on manual photo cropping procedures. However, it is worth considering the potential for developing an alternative neural network that can autonomously identify the eye inside an image, contingent upon the success of a proposed approach.

According to [22],Corneal ulcer is a prevalent ocular condition characterized by typical ophthalmic manifestations. Deep learning algorithms have been employed for the purpose of disease identification and quantification, utilizing a dataset consisting of 712 ocular pictures. The dataset comprises three distinct labels denoting the categorization of ulcers, namely generalizing, specifying, and signaling. Corneal ulcers are responsible for inducing a substantial infection within the corneal stroma or the epithelial layer of the eye. Fluorescein serves as the conventional diagnostic dye employed in the fields of optometry and ophthalmology. The utilization of machine learning methods in segmentation algorithms necessitates the use of extensive training datasets. The dataset encompasses a substantial collection of corneal ulcers, comprising photos of afflicted eyes that have been stained, along with corresponding fragmentation labels indicating the presence of flaky corneal ulcers. The classification standards for ulcer types and severity levels involve a total of five unique ulcer types and five levels of severity. These categories, known as the precise type-and-grading (TG) classification standards, are utilized to categorize the data. Corneal ulcers can be classified into many categories according to their different patterns and specific characteristics. The oval model developed for corneal extraction has the capability to partition the corneal surface into five distinct quadrants. The ocular staining images predominantly consist of type 2 and type 3, encompassing over 50% of the samples, with grade 5 being the most prevalent among the five categories. The segmentation labels on all 354 pictures have been visually reviewed by two ophthalmologists to verify accuracy. The dataset in question exhibits strong potential for the development of automated categorization systems, a subject of considerable scholarly attention.

In this paper [17], they provide an automated convolutional neural network technique for identifying corneal ulcer disease of the eye from a facial picture acquired by a general-purpose digital camera (CNN). In our proposed method, the eye area of the face is first identified and separated using Haar Cascade Classifiers. After that, CNN is used to assess whether or not there is a corneal disease. The experimental CNN model's original picture accuracy rate after using augmentation techniques was 99.43%, with sensitivity at 98.78% and specificity at 98.60%. Additionally, the specificity and sensitivity of the augmented photographs were 98.60% and 98.78%, respectively, with an average accuracy of 99.43%. Along with 287 photographs of corneal ulcers, they also use 187 images of other types of lesions. They use a number of augmentation methods, such as random rotation, horizontal and vertical shifting, inverting the horizontal and vertical axis, altering the lighting, adding salt and pepper noise, and random zooming. The system's first step is to capture a photo of the patient's face using a digital camera in order to check for any possible corneal ulcers. Four convolutional layers, a layer of max-pooling, and two fully-connected layers make up the design of CNN that we offer. For all layers, the activation function is the Rectified Linear Unit (ReLU) (ReLU). They devised an automated technique that utilizes face photographs to rebuild the human eye component. Using Haar Feature-based Cascade Classifiers, the system automatically detects faces and the ocular region. They employ the GrabCut method to remove the outer skin layer from the image of the eyes. In order to accurately remove the skin area and eyelid boundaries during the GrabCut segmentation technique, they employed a rectangular mask. Using the canny edge detection approach, the locations of the iris, upper, and lower eyelids are recognized. [17]

According to [27], The retina's capillaries are affected by a condition known as diabetic retinopathy (DR). If DR is identified early enough with routine health checks and treatment, it can be avoided. Computer vision and digital fundus retinal images were used to automatically identify DR at different stages. As another diagnostic tool, a model for the early diagnosis of diabetic retinopathy was developed. High memory and performance in F1 scores are seen in the APTOS data set for the non-healthy or DR-affected eye. To extract features from the network input, deep learning techniques are usually applied to classification tasks with a dense stack of convolutional layers. The absence of a standardized test data set prevents a fair comparison of our work with prior techniques.

From [23], we can summarize that the condition known as diabetic retinopathy (DR) damages the retina, causing long-term eye damage and occasionally catastrophic vision loss. When using DNNs, human interpretation can be significantly less necessary. The technique was tested using 400 retinal fundus pictures from the MESSIDOR database, which had a 97% accuracy rate. Before applying any machine learning technique, Principal Component Analysis (PCA) is used. It has the potential to identify high-dimensional data patterns in numerous applications, including finance, bioinformatics, data mining, biology, and psychology. A dimensionality reduction technique known as PCA can be used to extract the most important characteristics from a data set. Gray Wolf Optimization (GWO), a meta-heuristic algorithm that examines how grey wolves hunt, is one such method. The suggested model is built on a dataset of diabetic retinopathy. Principal component analysis (PCA) was utilized to eliminate or minimize less important components of the data.

In computer-aided settings, they used a deep learning algorithm for the physician-readers to make the facts and impacts of diabetic retinopathy understandable. The algorithm used for training the large dataset of retinal fundus images is Inception version-4 model architecture. They applied machine learning methods in settings involving computer-aided diagnostics. By analyzing the fundus images for the existence and severity of DR, they were able to calculate the sensitivity and specificity of each level of diabetic retinopathy. They also measured heatmaps on accurate values, speed tests, and the readers' level of confidence. The negative effect was seen in negative prediction because only integrating gradient methods was designed to show constructive prediction. [21]

(DR) Diabetic Retinopathy of detection using deep learning is based on images of four severity levels (for example - mild, moderate, severe no proliferative, and proliferative), where the Convolutional neural network (CNN) combines input images with a weight matrix to extract exact images without losing any specific information. Here, by finding pathological fundoscopy, a medical technique of visualization of the retina & detecting illness conditions using complex images of infected eyes, the whole working procedure was completed. Besides, fundoscopy image datasets were trained and retrained the models from the ImageNet database which is visual object recognition. Also, the final model was executed in TensorFlow by deep learning GPU interactive training system (DIGITS) and used a better version of GoogLeNet as a 2-ary, 3-ary, 4- ary classification model baseline. Again, the limitation of CNN, architectures were trained by a deep model called GoogLeNet. Moreover, for preprocessing, training, and data augmentation, images were transferred to a formation of hierarchical data format. Then, for training and testing models, DIGITS was used for data management, prototyping model & original time performance monitoring. On the contrary, they were not able to bring out significant sensitivity levels for the mild class because of lacking CNN. In conclusion, they achieved the greatest performance of CNN by using a binary classifier. This model allows CNN to diagnose non-diabetic diseases with high diversity and low bias.[13]

[8] The primary aim of the study conducted by Upadhyay, Karmacharya, Koirala, et al. is to determine the prevalence of ocular damage and corneal ulceration in the Bhaktapur district of the Kathmandu Valley. Additionally, the study aims to investigate the potential of topical antibiotic prophylaxis in preventing the development of ulceration following corneal abrasion. A longitudinal study was conducted from January 1992 to December 1993, in which a population of 34,902 individuals residing in the South Bhaktapur District was consistently monitored. The findings of this study provide compelling evidence for the timely application of 1% chloramphenicol ophthalmic ointment to individuals with corneal abrasions in rural areas as an effective measure for preventing corneal ulceration. The insights gained from this national intervention effort should prove highly valuable for other developing countries grappling with a high prevalence of corneal ulcers.

[30] The objective of the study conducted by Lv, Peng, Wang, Wu, et al. (2019) is to achieve precise categorization of corneal ulcers by the development of a multi-scale information fusion network utilizing a deep learning approach with label smoothing strategy. In this study, the proposed methodology initially employs the densely linked network (DenseNet121) as a fundamental framework for feature extraction. Furthermore, the multi-scale information fusion network (MIF-Net) was developed to enhance classification accuracy by effectively integrating shallow local information and deep global information through joint learning. The incorporation of the label smoothing technique is implemented in order to mitigate the influence of class similarity and intra-class variety on the feature representation. The performance of the proposed MIF-Net, which incorporates label smoothing, surpasses that of current state-of-the-art classification networks in terms of classification accuracy. Specifically, the weighted-average recall (WR) scores for the general ulcer pattern and specialized ulcer pattern achieve 87.07% and 83.84%, respectively. [28] The proposed methodology demonstrates promise in the classification of corneal ulcers observed in fluorescein-stained slit lamp images, hence facilitating ophthalmologists in achieving a more comprehensive and accurate diagnosis of corneal ulcers.

Chapter 3

Research Methodology

For the research, we have used 'The SUSTECH-SYSU dataset. [22] Then we completed data pre-processing, data analyzing, data augmentation, model building, applying deep learning algorithms etc. This part of our paper includes a description of the working process, the approach taken and data generation.

3.1 Approach

Initially, a comprehensive and diverse collection of photographs depicting corneal ulcers has been compiled. We ensured the inclusion of a diverse variety of ulcer manifestations and varying levels of severity within the dataset. We have used a dataset in our study, referred to as the "SUSTECH-SYSU dataset," which was acquired via Kaggle. The photographs within the collection are undergoing preprocessing via the use of consistent resizing techniques. The use of data augmentation technique has been employed to augment the dataset and improve the applicability of our model.

In order to provide annotations for each picture inside the collection, the appropriate categorization of corneal ulcer type or severity is applied to each image. The inclusion of this step is crucial in the development of the deep learning model and supervised learning. The dataset is partitioned into three distinct sets: the validation set, test set, and training set. The validation set has been used to optimize the model's hyperparameters and monitor its performance, whilst the test set is utilized to evaluate the ultimate correctness of our model. Next, an appropriate deep-learning-architecture is selected for photo classification. CNNs (Convolutional neural networks) were selected due to their ability to effectively capture hierarchical features from images, as well as their widespread use in many applications.

The neural network's architecture was constructed by the authors. In order to construct the selected Convolutional Neural Network (CNN) model, it is necessary to provide explicit details on the number of layers, filter sizes, activation functions, pooling methods, and any other architectural components. ResNet-50, VGG-16, and Inception-v3 are among the pre-trained models often used in various applications, each yielding distinct outcomes.

The deep learning model that was tailored to specific requirements underwent training using the designated training dataset. In the time of training part, the algorithm acquires the capability to recognise characteristics and patterns linked to different types of corneal ulcers. The loss function which we have used for multi-class classification is Softmax Loss (categorical cross-entropy). Adam has also had a position as an optimizer. The adjustment of hyperparameters was performed by using the validation set. Various learning rates, batch sizes, and regularization strategies have been used in order to increase the performance of the model.

3.2 Background Study

The research mainly focuses on finding the image types by using image classification. In this research, the data is image type. The images are divided into eye categories, types and grades. There are many research papers on this eye corneal ulcer disease. There are several customized models, hybrid models and pre-trained models by using different datasets. The trained and tested the models are done properly and got best validation accuracy, validation loss, training accuracy, testing accuracy etc. So, the research purpose and motive may be different but working on the topic is common. So, we reviewed some papers and found out their work, results and deficiencies. They used many common models but the difference is how they create, use and work on the model in every case. Then the work goes successfully. After the research, there will be future plans for the work. This is how these things work.

After reviewing many papers, we came across various research works on Corneal Ulcer while developing our procedure to get to the required outcome. There are several different types of classification models, including InceptionV3, VGG16, Sequential and ResNet50. In this section, we will provide a brief summary of the associated works we studied and some prior research on the algorithms and models that we desire to use in our research. Some background theories are incorporated into the models and algorithms we plan on utilizing, as well as some discussion of what other scholars have attempted in their own studies.

3.2.1 Image Classification

"Image classification" is a computer vision job that entails putting an input picture into one of several categories or labels depending on its content and feature set. It is the process of teaching a machine-learning model to point out and classify items or patterns in an image. There are many types of image classifications such as Binary Image Classification, Multi-Class Image Classification and multi-label Image Classification. The objective of binary image classification is to place each picture into one of two groups or classes. Identifying cats and dogs in photographs, as an example. The objective of multi-class image classification is to assign pictures to one of a number of predetermined classes or categories. Organizing fruit photos into groups like apples, bananas, and oranges is one approach. [25] [32] For example, A picture can concurrently belong to numerous classes in multi-label image classification. A photograph including both a cat and a dog, for instance, would receive labels.

Now, for doing image classification, there are some steps like collecting and preprocessing data, extracting features, selecting and training models, validation hyperparameter tuning and evaluation. First of all, it is necessary to assemble a database of photos with labels. These pictures have to be illustrative of the classes you intend to categorize. The preprocessing of the photographs may include reducing them to a prevailing size, levelling the pixel grades, and enhancing the data through rotation, cropping, and flipping procedures. Then, take pertinent information from the photos to enable the model to differentiate between various classes. This stage may use manually created feature extraction techniques in conventional computer vision, such as the Histogram of Oriented Gradients (HOG) or the Scale-Invariant Feature Transform (SIFT). In contrast, feature extraction in deep learning is automatically taught from the data. The next task is to select an appropriate machine learning or deep learning model for classifying images. The most popular models for this task are convolutional neural networks (CNNs), which can grasp hierarchical characteristics. Next, on the labelled dataset, to train the chosen model. By altering its internal parameters (weights and biases) in accordance with the training data and a predetermined loss function, the model learns to map picture attributes to class labels during training, followed by making sure the model generalizes well to new data and tests its performance on a different validation dataset. To ameliorate the rending of the model, it is required to adjust hyperparameters like learning rate, batch size, and model artifice. In the end, to determine the model's performance and accuracy with omitted data, it is important to evaluate it on a test dataset.

3.2.2 Model Training

Iteratively adjusting the model's parameters to enhance its performance on a particular task is a key step in machine learning and deep learning. As part of the procedure, training data must be gathered and prepared. A suitable model architecture, a suitable loss function, and an optimization technique must also be chosen. The training loop, which involves repeated repetitions of the training data, is the central component of the training process. The model gets the training data at the beginning of each epoch and then executes forward pass, loss calculation, backpropagation, parameter update, validation, early halting, termination, and evaluation. Apropos to verify the performance of the trained model on new data and ascertain its usefulness in practical applications, it is assessed on a separate test dataset. By repeating the training loop for multiple epochs and iteratively updating the model's parameters, the model learns to make accurate predictions and improve its performance on the specific task it was designed for.

3.3 Design the working flowchart

Here, in fig. 3.1 the Methodology workflow diagram is demonstrated.

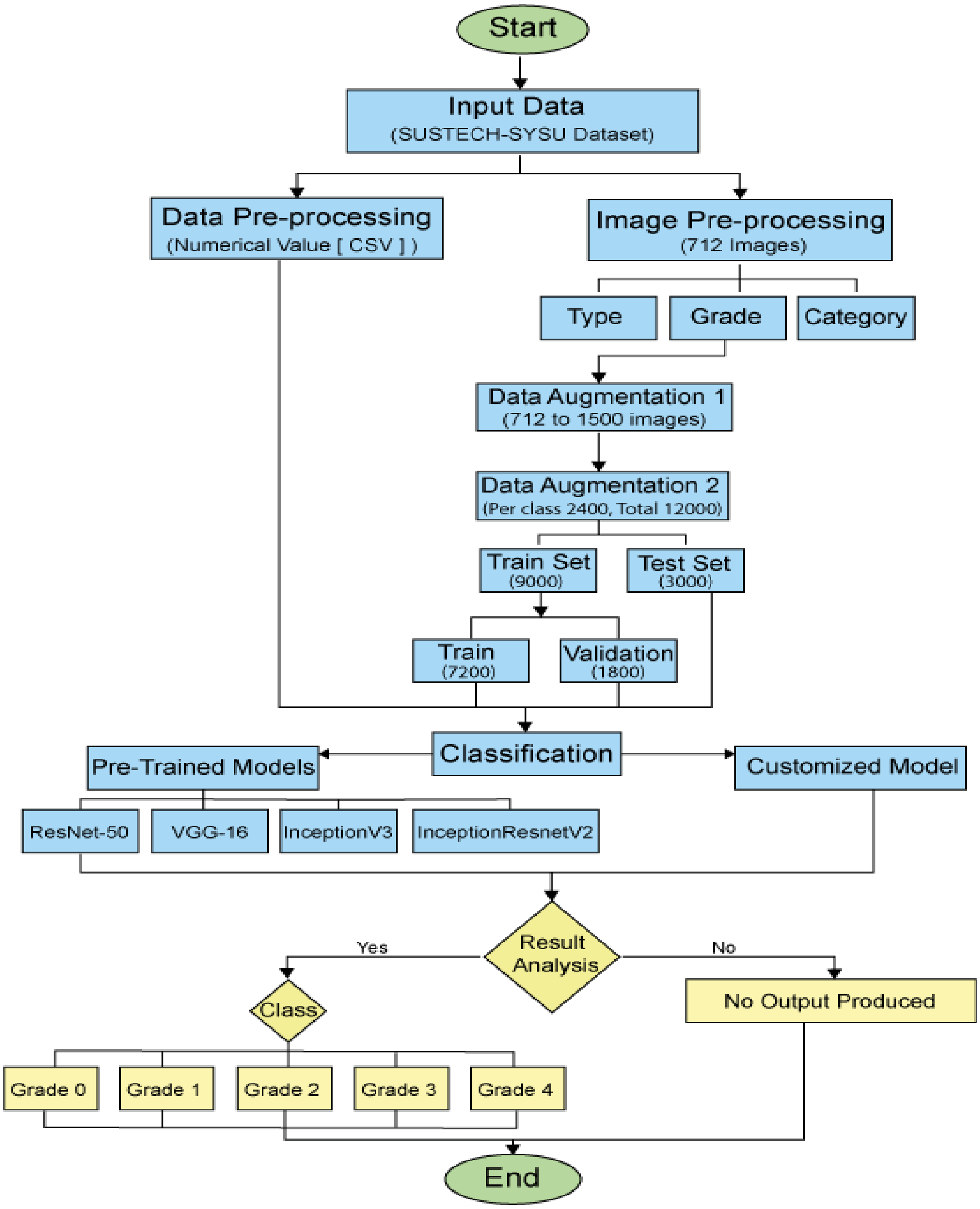


Figure 3.1: Working Flowchart

3.4 Data Generation

The dataset has significant importance within our research endeavor as it has a very crucial role in determining the validity of our study, since it is contingent upon both the data itself and the methodology used for its gathering. One of the primary concerns pertaining to our study is the lack of accessible datasets pertaining to the Electronic Control Unit (ECU). Furthermore, data collection by physical observation was not feasible in our study. In accordance with the task at hand, it was necessary for us to get a reliable dataset from online sources. Consequently, the "SUSTech-SYSU dataset" sourced from Kaggle was used for the purpose of our study.

Data generation flow given below -

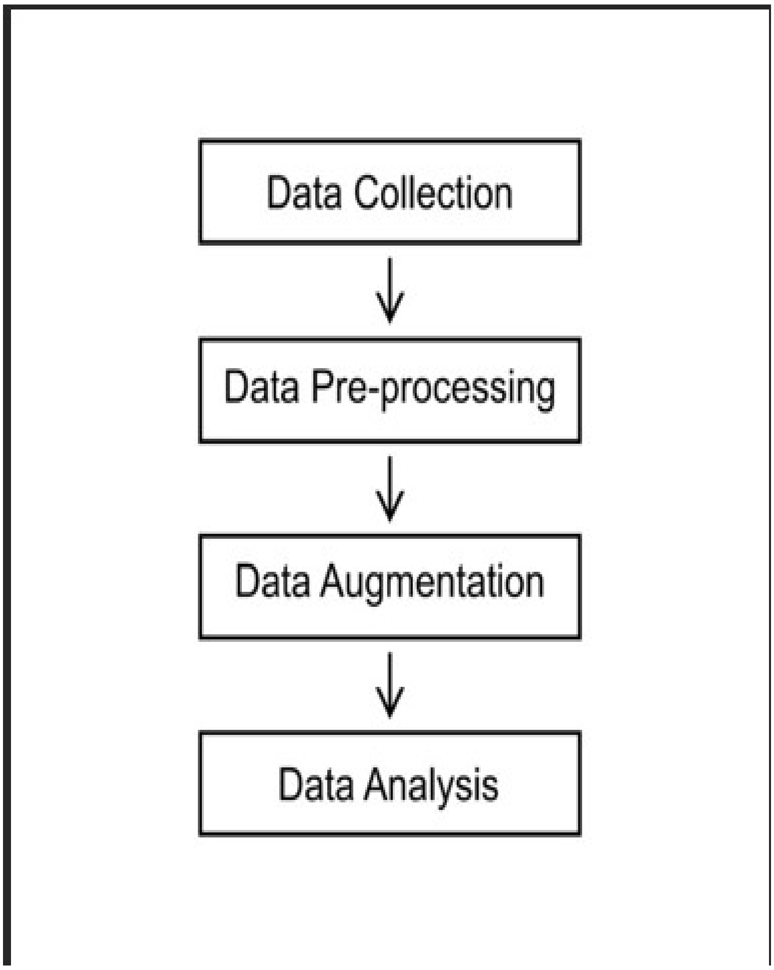


Figure 3.2: Data generation Flowchart

Chapter 4

Dataset

4.1 Dataset Collection

To date, there has been a lack of an openly accessible training dataset, importantly it is designed for the purpose of all supervised deep learning based segmentation of corneal ulcers. Additionally, in the appearance of an assessment dataset with accurate segmentation labels is crucial for the quantitative evaluation of various types of corneal ulcer segmentation methods. The "SUSTech-SYSU dataset" [5] was used for the purpose of autonomously segmenting and categorizing corneal ulcers, with the aim of delivering reliable diagnostic outcomes and producing pictures of superior quality. The collection comprises ocular staining photographs along with corresponding segmentation annotations depicting flaky eye corneal ulcer (ECU).

[22] At Sun Yat-sen University, The Zhongshan Ophthalmic Centre, amassed in total of 712 fluorescein staining photos of ocular surfaces. These eye images were obtained from patients with varying severity of eye ulcers. Variables such as age, sexual orientation, ulcer severity, and ulcer causation were not taken into consideration during the selection process. Every patient provided their informed permission. The research study technique obtained ethical clearance from the ethics council of the Zhongshan Ophthalmic Centre, an institution connected with Sun Yat-sen University in Guangzhou, China. The research was directed in adherence to the ethical standards outlined in the Declaration of Helsinki (2017KYPJ104).

[22] This experimental setup included the use of a white light source, which had the greatest extent at 30 mm. The light source was outfitted with a blue excitation filter and a diffusion lens. The diffusion lens was placed at an oblique angle spanning from 10 to 308 degrees. The light source was placed at a location midway between the border of the pupil and the limbus. Additionally, two magnification options, namely 10 and 16, were employed throughout the experiment. The eye hole, exposition duration, and shutter movement of an automated digital camera system are determined by analysing the level of light present in the examination room. The photos were acquired using a Canon EOS 20D digital camera and a Haag-Streit BM 900 slit-lamp microscope, both produced by Haag Streit AG in Bern, Switzerland.

4.2 Dataset Background

Corneal ulcers are the prevailing manifestation of Corneal Disease. An eye corneal ulcer (ECU) refers to a distressful or in many severe cases, an infectious ailment that results in harm to the stroma or epithelium of the cornea. Numerous segmentation algorithms heavily depend on machine learning methodologies, necessitating extensive training datasets. An image segmentation training sample comprises the picture that requires segmentation and the corresponding ground truth segmentation label. The efficiency of a machine learning-based segmentation system is often significantly influenced by the amount and variety of the training dataset. The study's focus on the production of extensive and high-quality training datasets has gained significance within the domain of medical picture segmentation. The use of large-scale photographs is essential within the context of our thesis, which focuses on the field of medicine. The images depict ocular corneal ulcers. In order to effectively train our CNN model-based technique, it is essential to have access to a substantial collection of eye images accompanied by accurate labelling, including various illnesses, types, and classifications.

4.3 Data Analysis

Data analysis is the systematic process of examining, refining, manipulating, and evaluating data with the objective of identifying recurring patterns, extracting significant insights, and formulating informed judgements. The extraction of useful information from raw data is a crucial step in the data science process, which finds application in many domains and industries. Various approaches and procedures are used in the analysis of data in order to get a deeper comprehension of the data and its inherent patterns. Data analysis may be characterised by many fundamental phases. Data collecting, data purification, data exploration, data transformation, data analysis methodologies, pattern discovery, data visualisation, data validation, testing, reporting, and communication are some of the key processes involved in the field of data analysis.

We have read the data using pandas. We reformat data for machines and maintain consistency for the image data. Then we used cleansing of data to rectifying our data where there were no missing data and check the appropriateness of the data but our data was perfect in these cases. So we moved to data compression. he frequency of Category type and grade classes are given above by data analysis.

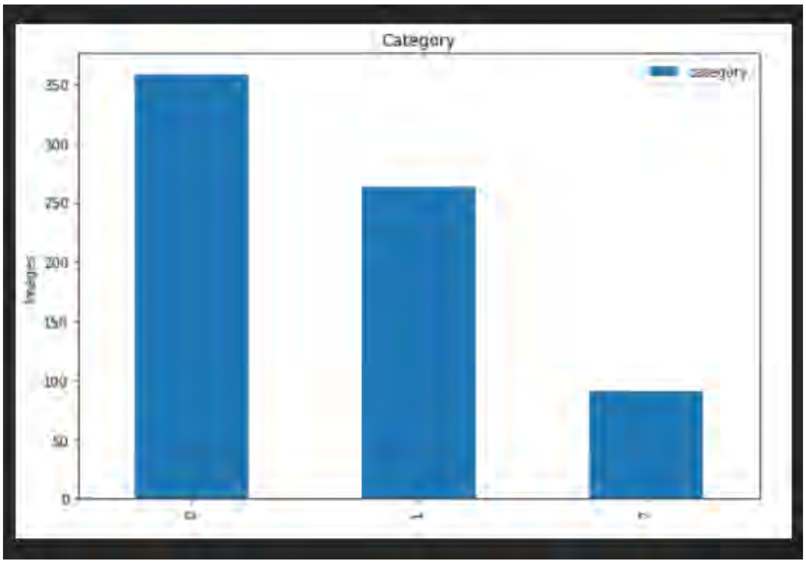


Figure 4.1: Category class Data Analysis

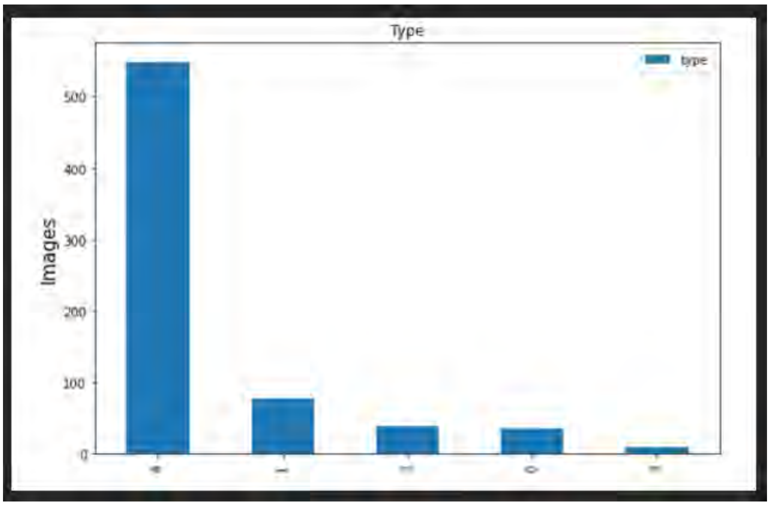


Figure 4.2: Type class Data Analysis

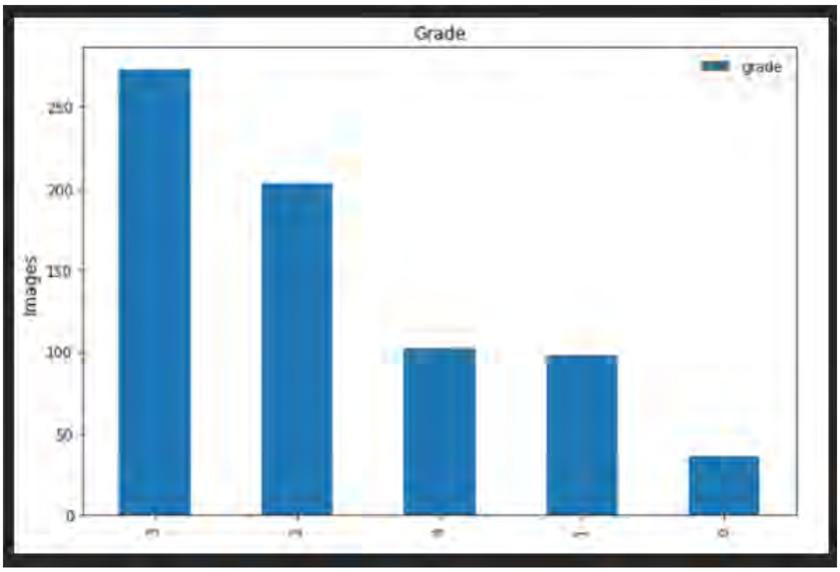


Figure 4.3: Grade class Data Analysis

4.4 Image Description

Each picture in the collection is associated with three class labels. Primarily, each image is accompanied by a caption describing the comprehensive ulcer pattern. Furthermore, each image is appropriately annotated based on its unique ulcer pattern. Furthermore, it is important to note that each image is accompanied by a descriptive description that indicates the level of severity of the ulcer.

[22] The dataset has three separate categories, namely point-like corneal ulcer, point-flaky mixed corneal ulcer, and flaky corneal ulcer. In addition, it is important to acknowledge that the ground truth segmentation labels include flaky and flaky corneal ulcers while omitting point-like ulcers from the scope of the investigation. The dataset has been categorized into five distinct categories, with each category representing a unique ulcer pattern. Moreover, the provided information encompasses five distinct categories that assess the severity of ulcers, using specific type-and-grading classification criteria.

4.4.1 Image Category

Ulcers may be classified into three basic groups based on their overall pattern. The corneal ulcer may be classified into three distinct varieties, which are determined by the shape and distribution of their features. [22] The classifications include the point-like corneal ulcer, the point-flaky mixed corneal ulcer, and the flaky corneal ulcer. The items in question are assigned numerical values, namely 0, 1, and 2. A corneal ulcer that appears as a little point is indicative of the most prevalent and least severe kind of corneal infection. Point-like corneal ulcers often manifest either at the advanced stages of ulcer healing or at the onset of an eye infection. Corneal ulcers often manifest as a conglomeration of several tiny ulcerative lesions and have the potential to emerge at any location on the cornea. Typically, individuals experience recovery via the use of effective and prompt therapeutic interventions. For Category class the graph is

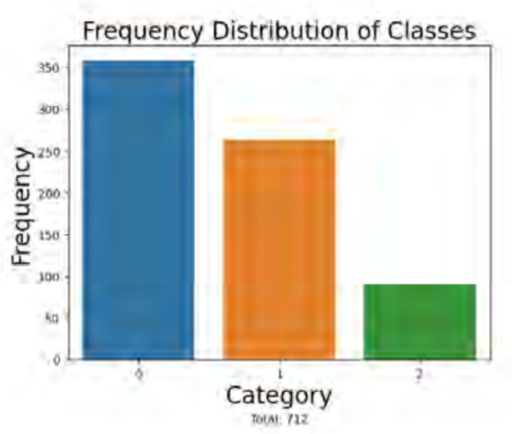


Figure 4.4: Catergory Class Graph

A corneal ulcer with a crumbly texture, characterised by a profound green colouration and well-defined margins, serves as an indicator of a possibly life-threatening corneal ailment. Individuals who present with a corneal ulcer characterised by the presence of flakiness may experience the formation of scars on their ocular surface, which may exhibit varied degrees of thickness. These scars have the potential to cause substantial impairment to visual acuity and, in severe cases, may lead to complete loss of eyesight. A point-flaky mixed corneal ulcer refers to a corneal ulcer that exhibits uneven distribution characterised by the presence of both flaky and point-like lesions. [5] The presence of a corneal ulcer displaying a point-flaky mixed pattern often indicates the presence of a corneal disease that falls within an intermediate range of severity, positioned between the two groups stated before.

[22] The whole dataset was dispersed in the following manner: Out of the whole dataset, it was determined that 91 photographs had flaky corneal ulcers, while 358 images displayed a combination of point and flaky ulcers. Additionally, an additional 358 photos were recognized as depicting corneal ulcers characterized by the presence of flaky tissue. Figure 4.1 illustrates instances representing each of the three distinct categories. Examples of each of the three kinds are shown in Fig. 4.5.

The upper row consists of corneal ulcers characterised by a point-like appearance, the middle row has corneal ulcers exhibiting a combination of point-like and flaky characteristics, and the lower row is composed of corneal ulcers mostly displaying a flaky morphology. The diagram depicts the three discrete categories of corneal ulcers and their respective underlying patterns.

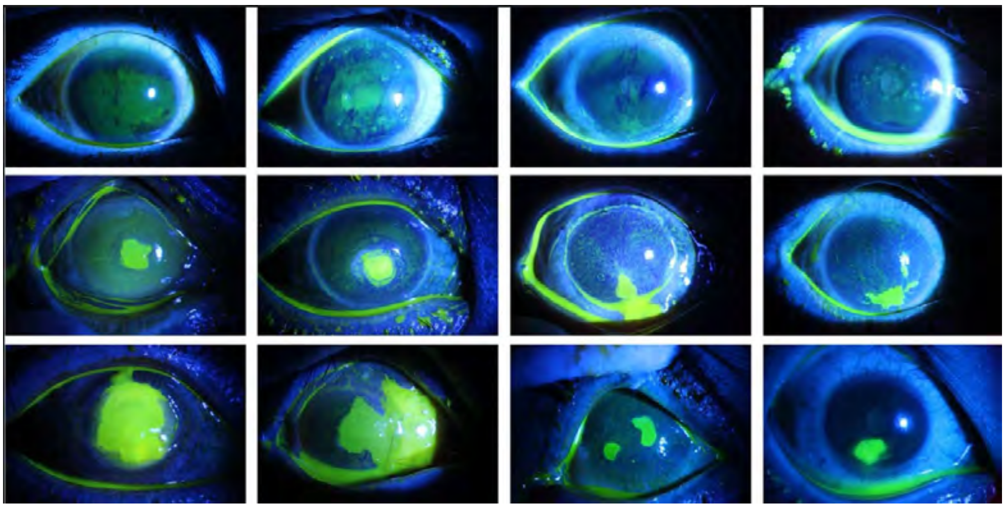


Figure 4.5: Eye Image Catergory

4.4.2 Image Type Grading

Various types of corneal ulcers exist, characterised by various patterns. Identifying the specific kind serves as the first diagnostic step for physicians to ascertain the likely underlying condition. The specific kind of condition has an impact on the corresponding course of therapy as well. The photos were categorised into five distinct categories. The numerical values shown are 0, 1, 2, 3, and 4. The observed frequency distributions consist of 36 photographs for type 0, 98 pictures for type 1, 203 pictures for type 2, 273 pictures for type 3, and 102 pictures for type 4.

The grading standards for the different types are shown in Figure 4.6.

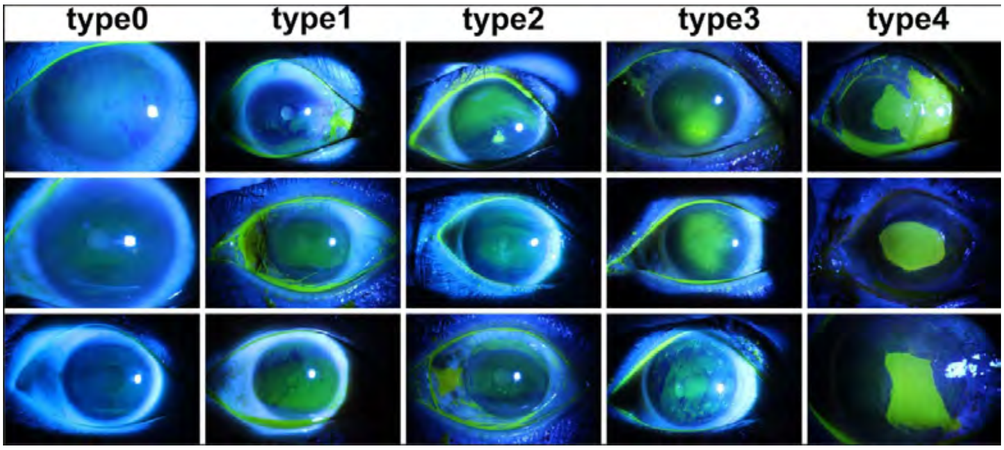


Figure 4.6: Image Type Grading

During the preliminary phase of our study, an in-depth analysis was performed on our dataset, including several attributes such as the "type," "grade," and 'category' associated with each image. In order to obtain a comprehensive understanding of the distribution patterns within these categories, we employed the use of graphical bar charts to visually depict the distribution of values within each respective category. The aforementioned investigation had a crucial role in comprehending the composition of the dataset. For the type class:

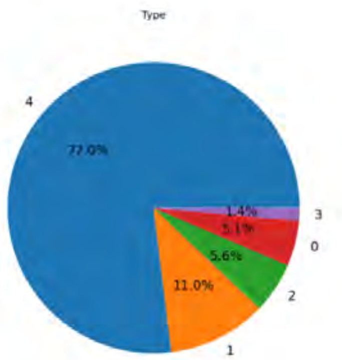


Figure 4.7: Type Class Pie-Chart

4.4.3 Image Grade Grading

[22] There exists a substantial correlation between the particular location of a corneal ulcer inside the cornea and the degree of severity it exhibits. The pupil, a minute

circular opening located in the middle part of the cornea, enables the transmission of light into the eye. The destruction of the epithelium near the pupil significantly affects visual perception. The presence of an ulcer in the pupil area results in the obstruction of light entry and consequent impairment of vision. In some cases, it is possible for ulcers to have a tendency to propagate over the course of treatment, therefore signifying a deterioration in the corneal infection. The current medicinal treatment is ineffective in halting bacterial multiplication and needs immediate modification. Based on the elliptical model previously provided for corneal extraction, it is suggested that the spatial data about the corneal ulcer may be partitioned into five quadrants. The grading system is structured with numerical values ranging from 0 to 4. The grade 0 signifies the absence of a corneal epithelial ulcer. Grade 1 corneal ulcers have localised effects limited to a single quadrant of the surrounding area. A corneal ulcer classified as Grade 2 involves the involvement of two adjacent quadrants. Grade 3 ulcers include three of the four neighbouring quadrants. In summary, it can be said that a grade 4 ulcer mostly impacts the central optical zone of the cornea.

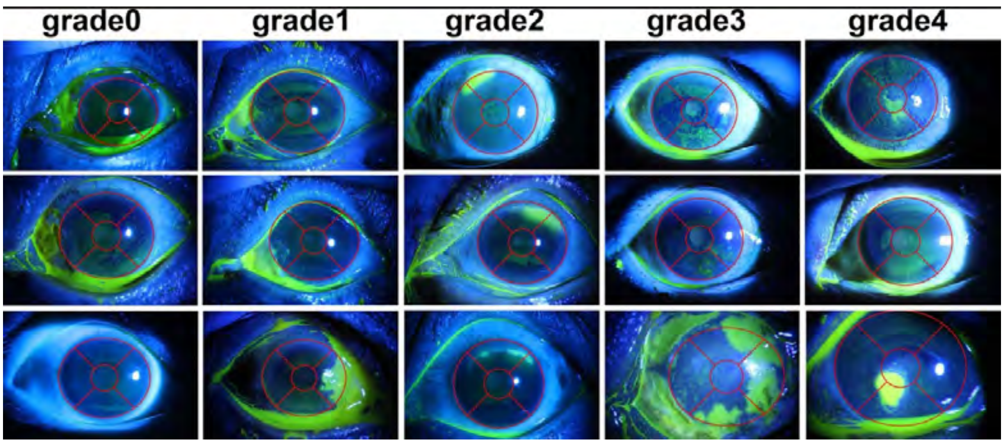


Figure 4.8: Image Grade Grading

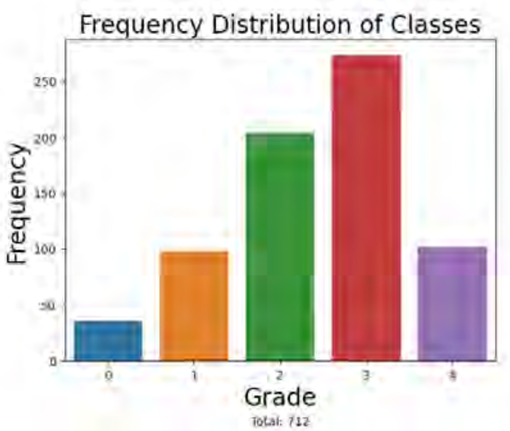


Figure 4.9: Grade Class Graph

A majority of our ocular staining photographs exhibit a grade 5 classification according to the five-tier grading system. The distributions may be described as follows: There are 36 photographs available for students in grade 0, 78 pictures for students in grade 1, 40 images for students in grade 2, 10 images for students in grade 3, and 548 images for students in grade 4. The grading rules for Grade are shown in Figure 4.9.

4.4.4 Image label Creation

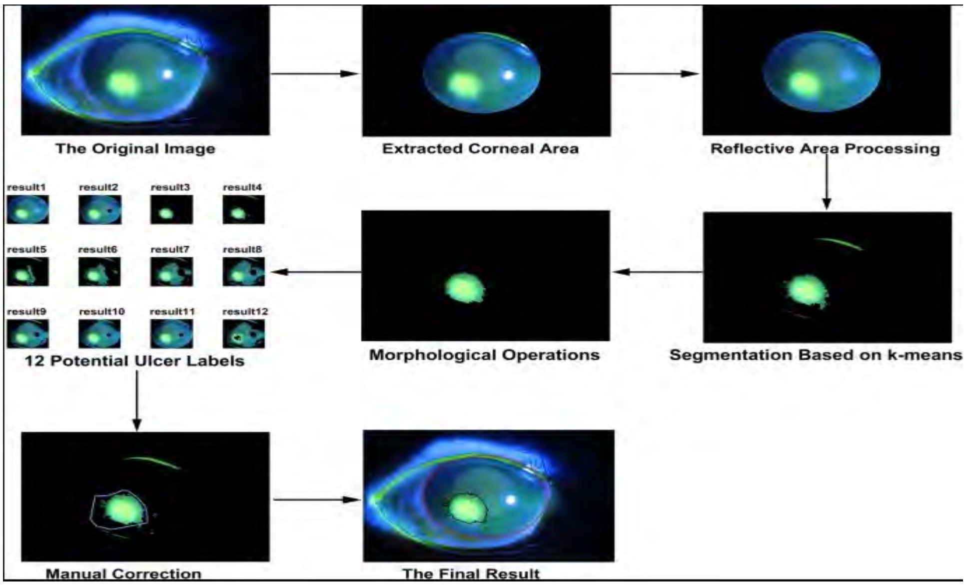


Figure 4.10: Image label Creation

The task of manually differentiating the sites of corneal ulcers that have a point-like appearance is a significant challenge. Consequently, while dealing with flaky ulcers, it is necessary to only generate point-flaky mixed ulcer pictures and flaky ulcer images that are accompanied by accurate segmentation labels. The cornea was originally identified in each picture via the use of a semi-automatic methodology. The primary ulcer region was subsequently delineated within the cornea through the utilisation of several methodologies. [22] These methodologies encompassed: 1) the modification of colour information pertaining to reflective regions in the initial fluorescein staining image of interest; 2) the application of iterative k-means-based clustering to extract regions exhibiting similar colour characteristics; and 3) the implementation of morphological operations to refine the outcomes of the preceding step. The morphological operators' parameters were computed automatically using linear regressors. Existing literature extensively examines each stage in great detail. After the original ulcer site was removed, an additional 11 ulcer labels were generated, as seen in Figure 4.4, in order to provide more options for eventual human correction.

4.5 Dataset Augmentation

Data augmentation is a commonly used technique within the domain of machine learning, particularly in relation to computer vision applications. The main aim of this approach is to augment the diversity of the training data by producing altered iterations of the initial dataset. The process involves implementing a range of modifications or adaptations to preexisting data in order to produce novel samples that include similarities to, although are not exact replicas of, the original data. Data augmentation is particularly advantageous in scenarios when the training data set is constrained in size, as it serves to mitigate the issue of over-fitting and enhances the overall capacity of the machine learning model to generalise. In addition to performing flipping and rotating operations, the introduction of diverse augmentations, such as translation, scaling, and random transformations, may be used to increase the variability of the training dataset. Data augmentation is a widely used technique in computer vision applications that rely on deep learning, including but not limited to picture classification, object recognition, and semantic segmentation. The selection of augmentation methods and their corresponding parameters is contingent upon the dataset's quality and the requirements of the task at hand.

1. **First Augmentation: Custom Image Augmentation**
   In order to solve the problem of class imbalance present in the dataset, a customized image augmentation approach was implemented. The implemented approach includes generating an additional batch of 300 photos for each class in order to get a balanced dataset. Every augmented image saved its original dimensions of 2592 pixels in width and 1728 pixels in height. A Python function, named ImageAugmentation, was built to apply a variety of augmentation approaches randomly in order to enhance the dataset. The implemented techniques featured horizontal and vertical flips, random rotations, and mirroring. The previously mentioned method had a key role in attaining a well-balanced dataset through the augmentation of minority class representation. Here is the graph for the first augmentation.

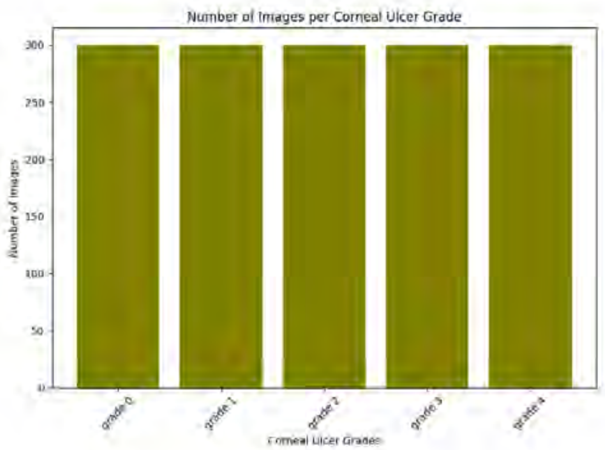


Figure 4.11: 1st Data Augmentation Balancing

2. **Second Augmentation: Automated Augmentation**
   In order to get a balanced representation of all classes inside our augmented dataset, we implemented class-wise augmentation. The procedure mentioned earlier was executed for every category of "grade" (Grade 0, Grade 1, Grade 2, Grade 3, and Grade 4). The objective of the augmentation was to achieve an evenly distributed distribution through ensuring an equal count of 2400 photos for each class.

   In relation to each individual class: The quantity of samples contained in that class was determined. The process of augmentation was carried out utilizing the ImageDataGenerator that had been established previously. The images that were produced had been saved in directories that corresponded to their respective classes. The process of augmentation was carried out until the required number of samples (2400) was achieved for each class.

   The composition of the dataset: By implementing robust augmentation techniques, we effectively achieved stability within our dataset, producing an overall count of 12,000 images. In order to ensure equal representation, a total of 2,400 photos were included in each of the 5 classes. The earlier mentioned balanced dataset served as the fundamental basis for our following processes of model training and evaluation. Here is the graph for the final augmentation:

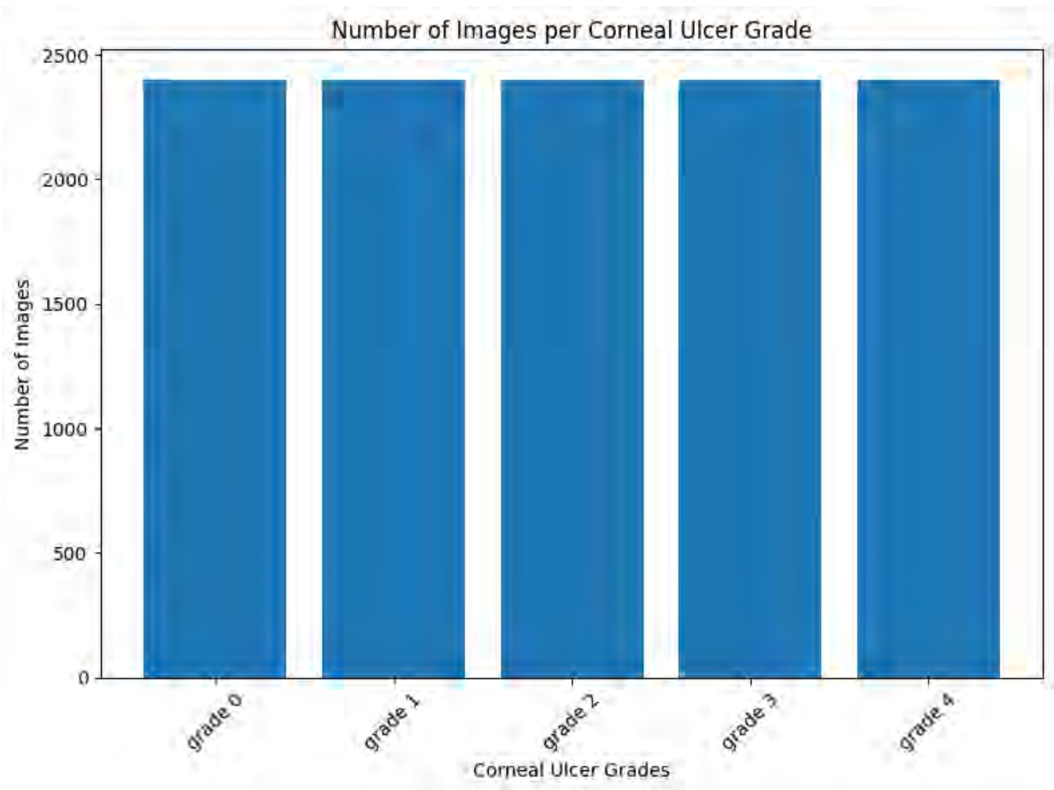


Figure 4.12: 2nd Data Augmentation Balancing

Expanding upon our previous augmentation tasks, we applied the reliable Keras ImageDataGenerator to enhance the dataset using further augmentation techniques. This particular stage contained a variety of transformations:

Rescaling: The values of each pixel were set to be between 0 and 1.
Rotation: The images underwent random rotation with a maximum deviation of 20 degrees.
Horizontal and Vertical Shifts: Shifts were made randomly in both the horizontal and vertical directions, giving the positions of the objects more variety.
Shearing: In order to create unpredictability, random shearing transformations were utilized.
Zooming: Images were zoomed in and out at random to mimic different sizes of images.
Horizontal Flipping: Some of the pictures were turned horizontally at random.

The result of data balancing after data augmentation 2 has already shown in fig 4.12.

4.6 Description of Data-Processing

Data preprocessing refers to the systematic transformation of unprocessed data into a structured format that may be effectively interpreted and analysed by computers and machine learning algorithms.Improper training of machine learning models utilising datasets might lead to the development of models that are rendered ineffective for analytical purposes. The significance of clean and preprocessed data surpasses that of the most advanced algorithms. In fact, Machine learning (ML) models that are trained on dirty data has the capacity to impede the desired analysis. Data that has undergone thorough preprocessing and cleansing will be more suited for subsequent operations, leading to increased precision. The concept of "data-driven decision-making" is often emphasised, yet it is important to acknowledge that judgements based on erroneous data may nonetheless result in unfavourable outcomes.

The attributes of a data collection, sometimes referred to as "features," serve the purpose of characterising or expressing the dataset. The determination of this may be influenced by several aspects, including but not limited to size, location, age, time, colour, and other relevant factors. The phrases "attributes," "variables," "fields," and "characteristics" are all used to denote various aspects. In datasets, variables are represented as columns. In the process of data preprocessing, it is crucial to possess a comprehensive understanding of the concept of "features," since the selection and prioritisation of these characteristics by the organisation are contingent upon its specific objectives. First, let us examine the two main sorts of attributes used to classify data—categorical and numerical:

Categorical features: Categorical features refer to features that derive their justifications or values from a predefined set of possible reasons or values. Categorical values include a range of attributes, such as the colours assigned to houses, various animal species, the months of a year, binary distinctions denoted by True or False, as well as the classification of sentiments into positive, negative, and neutral categories, among others. The attributes may be categorised into many categories.

Numerical features: Numerical aspects are defined as variables that have values on a scale, demonstrating statistical, integer-related, or continuous attributes. Numerical quantities may be represented using whole integers, fractions, and percentages. Numerical components include many quantitative aspects, such as the pricing of residential properties, the word counts inside a written piece, the durations of journeys, and other such numerical variables. Let us examine the sequential procedures that are necessary to validate the adequacy of data preprocessing.

Let's look at the established procedures that must follow to ensure the data has been correctly preprocessed.

4.6.1 Data quality evaluation:

When data is collected from several sources, it may be obtained via various means. The primary objective of this approach is to reformat data for machines, but it requires the initial availability of well prepared data. The presence of mixed data values may indicate that several feature descriptors, such as "man" or "male," were used by a significant number of sources. Consistency should be maintained in all of these value descriptions. The presence of data outliers may significantly influence the outcomes of data analysis. Conduct a thorough examination to identify any instances of missing data, vacant text boxes, incomplete data fields, or unanswered survey questions. The potential cause of this issue might be attributed to inaccurate or incomplete data.

4.6.2 Cleansing of data:

Data cleaning, also known as data cleansing, refers to the systematic procedure of rectifying missing data and eliminating inaccurate or unnecessary material from a dataset. Data cleaning is considered to be the foremost preprocessing step since it plays a crucial role in preparing the data for subsequent processes.

4.6.3 Data Balancing:

Data balancing refers to the procedure of decreasing the size of a file or dataset with the objective of conserving storage capacity, enhancing the efficiency of data transfer, and expediting processing. The process entails the balancing of data by using a reduced number of bits compared to the initial representation, while ensuring the preservation of crucial information and retaining correctness. Data balancing methods are extensively used in several domains, including computer science, telecommunications, multimedia, and data storage.

4.7 Description of Image Processing

[3] The categorization of image pre-processing algorithms may be based on the size of the pixel neighbourhood used for the computation of the brightness of a new pixel, resulting in four distinct groups. The process of preprocessing involves making adjustments to the brightness of pixels, exploring geometric changes, using preprocessing algorithms that analyse the immediate area around the targeted pixel, and briefly touching upon picture restoration, which A thorough comprehension of the whole of the picture is necessary. Image pre-processing methods use the substantial redundancy present in images. In authentic photographic images, adjacent pixels that pertain to the same object have brightness values that are closely similar or comparable. Image preprocessing techniques aim to standardise, enhance, and reduce noise in images, hence optimising their performance in subsequent tasks. Commonly used methods for preparing images encompass:

4.7.1 Rotation and Flipping

Rotation and flipping are often used data augmentation methods in computer vision applications. These approaches aim to enhance the diversity of training data and enhance the generalizability of machine learning models. The process of creating a mirror image by horizontally flipping a picture is a valuable technique in scenarios when the object's orientation is nonessential. [15] Images depicting various item orientations are generated by the process of rotating an image by a certain angle. Consequently, the inclusion of rotations or horizontal/vertical flips has the potential to expand the dataset and enhance the model's capacity for generalisation.

The augmentation techniques used to the training data, such as rotation and flipping, enhance the model's ability to withstand variations in orientation. This augmentation also aids in preventing overfitting to certain orientations that may be present in the original dataset. The act of rotating and flipping data results in an expansion of the training dataset, so providing the model with a greater variety of instances to acquire knowledge from. Nevertheless, it is important to use these strategies judiciously and take into account the inherent characteristics of the issue at hand. Excessive rotation or flipping might potentially introduce noise and negatively impact the performance of the model in tasks that involve object orientation.

4.7.2 Data Resizing and Train- Test Split

Prior to the training of the model, we went through image resizing to achieve a standard resolution of 224 x 224 pixels. The act of scaling not only enabled effective training of the model, but also ensured its compatibility with widely used deep learning architectures. In order to assess the efficacy of our model, we split the dataset into separate training and testing sets. Around 25% of the dataset was set aside for the purpose of testing, while the remaining 75% (equivalent to 9,000 photos) were allocated for training. The implementation of this division permitted an in-depth analysis of the model's capacity to efficiently generalize and accurately classify corneal ulcer grades.

4.7.3 Normalization

Normalisation is a crucial step in data processing, whereby the numerical values of characteristics in a given dataset are adjusted to a standardised range. The objective of normalization is to provide a consistent level for all characteristics, hence reducing the likelihood of any one feature exerting disproportionate dominance or impact on the machine learning (ML) algorithm's learning process. The process of normalisation has particular importance in the context of distance-based algorithms or gradient-based optimisation techniques, as the possibility exists for the enhancement of model convergence and stability throughout the training period.

Normalisation is a technique used to guarantee that all features are assigned equal importance throughout the learning process, irrespective of their initial scales or units. In order to ensure consistency, it is recommended that normalisation be applied just to the training data. Subsequently, the validation and test datasets should be subjected to normalisation using identical parameters, such as minimum, maximum, mean, and standard deviation. [34]

4.7.4 ImageDataGenerator

The ImageDataGenerator class is a commonly used component of the Tensorflow Keras API, specifically designed for the purpose of augmenting data during the training process of Convolutional Neural Network (CNN) deep learning models. The use of the Imagedatagenerator class facilitates the process of augmenting image data. Data augmentation is the introduction of random modifications to the training data in order to enhance diversity, hence improving the model's ability to generalise and withstand variations. Data augmentation procedures are often used while using the [10] ImageDataGenerator. The transformations include a variety of picture manipulations, such as rotation, alterations in width and height, shear, image zooming, horizontal and vertical flipping, modifications to brightness and contrast, normalization, and other analogous operations. The Keras ImageDataGenerator is used to accept the input of the original data and afterwards apply random transformations to it, resulting in an output that only consists of the modified data. The class known as the image data generator encompasses several methods and parameters that aid in the determination of the behaviour associated with producing data. The ImageDataGenerator class is used by providing appropriate parameters and necessary input to it. [33]

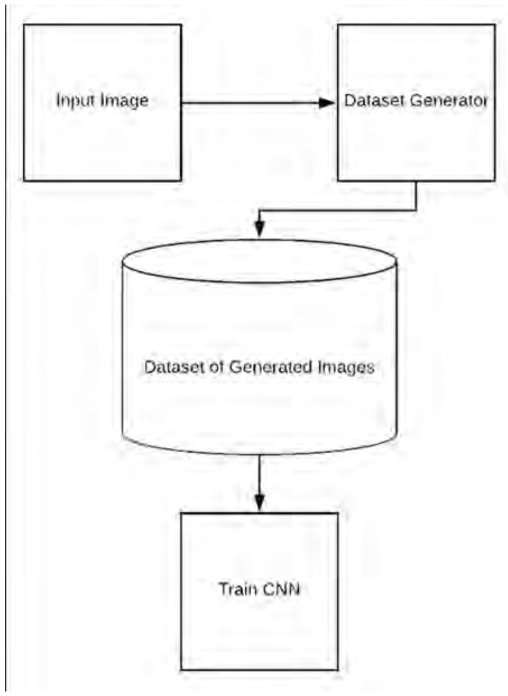


Figure 4.13: ImageDataGenerator sequence

During this stage of the process, the original image data is retrieved from the disc. Subsequently, the image undergoes a random modification process including a sequence of random translations, rotations, and other similar operations, before being saved back into the storage medium. Execute this procedure n times. Consequently, the small dataset will undergo a transformation, resulting in a substantial increase in size. Subsequently, it is important to begin the training process for the model.

Chapter 5

Model Selection

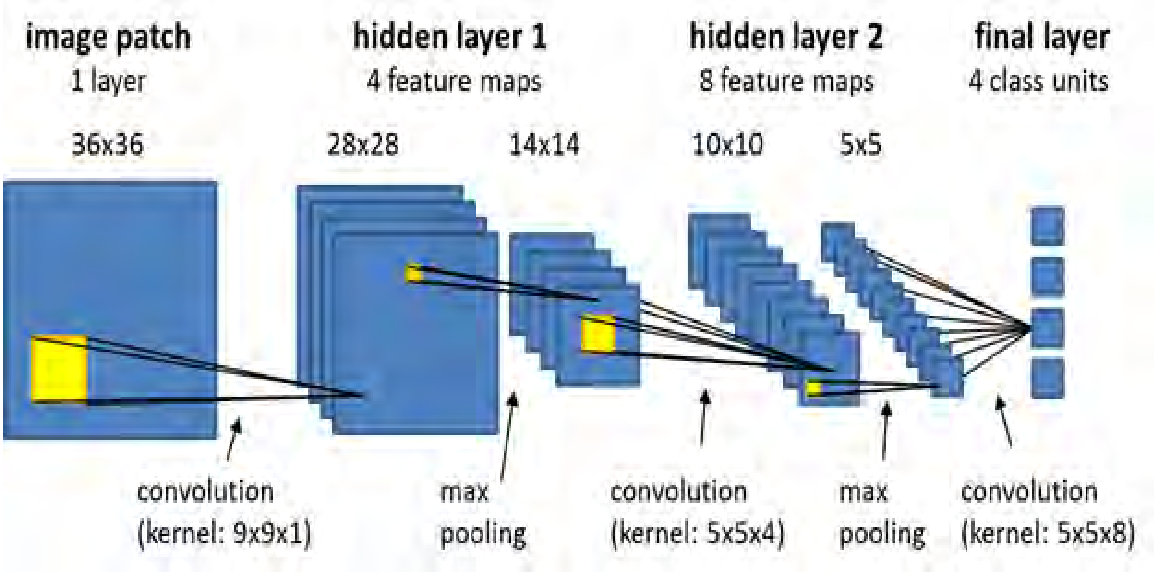
5.1 Pre-trained Model

A machine learning model that has been pre-trained has been used by someone else or a third party organization and has been trained on a sizable dataset. These models are developed employing robust hardware and computing resources and trained on large datasets, which is sometimes difficult and time-consuming for lone developers or academics to reproduce. Pre-trained models are useful tools for a variety of tasks in computer vision, natural language processing, and other fields because they have already learnt meaningful representations, features, and patterns from the training data.
[16] It can be time- and resource-consuming to train a deep learning model, such as a Convolutional Neural Network (CNN) for image recognition or a Transformer for natural language processing. A huge and varied dataset is fed to a model during pre-training, and the model then learns to identify pertinent patterns and relationships contained in the data.
The weights and parameters of the trained model are stored and made accessible for download when the pre-training is finished. For certain tasks or domains, these pre-trained models can then be applied as a foundation or feature extractor. [29] The process of fine-tuning, which is frequently used with pre-trained models, involves further training the model using a smaller dataset that is frequently unique to the intended job. This saves time and computing resources by enabling the model to modify its previously learnt representations to the new job rather than having to start from scratch.
Transfer learning, enhanced performance, shorter training times, and community accessibility are benefits of employing pre-trained models. Popular pre-trained models include language models like BERT, GPT, and RoBERTa as well as ImageNet pre-trained CNNs like VGG16, ResNet50, and Inception v3. Overall, pre-trained models have, in general, transformed the area of deep learning by enabling the quick creation of complex models and democratizing access to cutting-edge methods. [6]

5.1.1 Convolutional Neural Network (CNN)

CNNs, or Convolutional Neural Networks, are a sort of neural network optimized for scheming data with a grid-like design, such as an image. The visual statistics is altered into binary in a digital image. It's made up of a mesh of pixels, each of which has a numerical value that specifies its brightness and colour.

CNN Architecture of Layers' Workflow [18] :



Convolutional Neural Network (CNN) is frequently employed for picture segmentation and classification tasks including eye corneal ulcer (ECU) detection and classification. In terms of automated image processing and classification, convolutional neural networks (CNNs) are now considered to be the best algorithms available. Different CNN models are better suited for certain tasks. It has found particularly widespread application in the medical field, particularly in the analysis of X-ray and MRI scans, and corneal ulcers.[18]

Formulas for CNN operation [35] :

[35] In the formula $z^l = h^{(l-1)} * W^l, z^l$ glosses the activation map or feature map in the l-th layer of the CNN. Each layer in a CNN typically consists of multiple feature maps, but this formula represents the computation for a single feature map. $h^{(l-1)}$ represents the activation map or feature map(output) from the previous convolutional layer, l-1. In a CNN, the activations in each layer are computed by applying a convolution operation to the preceding layer's feature map, followed by an activation function. W^l is the weight tensor or kernel for the l-th layer's convolutional operation. Each feature map in a CNN has its own set of weights or kernels. These weights are learned during the training process, and they determine the specific features that the feature map will detect.

[35] The convolution operation is represented by the '*' symbol in the formula. It involves taking the convolution of the previous layer's feature map $h^{(l-1)}$ with the kernel W^l in the l-th layer.The result is a new feature map z^l that represents the presence of specific features or patterns detected by the convolutional layer. This process is fundamental to CNNs, as it allows the network to automatically learn and extract hierarchical features from the input data, enabling it to recognize complex patterns and objects in images or other types of data. The activation function is typically applied after this convolution operation to bring in non-linearity and amplify the network's aptitude to capture shrewd patterns.

[35] $h^{(l)}{}_{xy} = max_{(i=0...s,j=0...s)}, h^{(l-1)}(x+i)(y+j)$, this formula represents the computation of a single element $h^{(l)}{}_{xy}$ in the $(l-th)$ layer of a Convolutional Neural Network (CNN) using the max-pooling operation. $h^{(l)}{}_{xy}$ is the value at position (x,y) in the feature map of the $(l-th)$ layer. It results from applying max-pooling to the feature map from the previous layer, $h^{(l-1)}$. In simpler terms, $h^{(l)}{}_{xy}$ represents the output of the max-pooling operation at position (x,y) in the l-th layer. $h^{(l-1)}(x+i)(y+j)$ refers to the values in the feature map of the (l-1)-th layer, specifically within a local region defined by indices (x+i) and (y+j). Here, i and j iterate over a window of size sxs centered around position (x,y) in the (l-1)-th layer's feature map.

[35] $max_{(i=0...s,j=0...s)}$, represents the maximum operation applied to all values $h^{(l-1)}(x+i)(y+j)$ within the local region defined by the window. It calculates the maximum value from this region. In essence, max-pooling is a down-sampling operation used in CNNs to reduce the spatial dimensions of feature maps while preserving important information. The formula essentially finds the maximum value within a local neighbourhood that is defined by the window of size sxs in the (l-1)-th layer's feature map and assigns that maximum value to position (x,y) in the l-th layer's feature map. This process is repeated across the entire feature map, effectively reducing its spatial dimensions by a factor of s which is 2 generally in each dimension. The outcome is a down-sampled feature map with a reduced spatial resolution, but it retains the most significant features, which is valuable for tasks like object recognition where preserving critical patterns while reducing computation is essential.

[35] In the formula of CNN operation for a fully connected layer, $z_l = W_l * h_{(l-1)}, z_l$ This represents the output of the fully connected layer, l. Each layer in a CNN typically consists of multiple neurons, and z_l is a vector that contains the activations of these neurons. These activations are computed as a weighted sum of the activations from the previous layer $h_{(l-1)}$. W_l is the weight matrix specific to the fully connected layer l. Each row of W_l corresponds to the weights associated with a particular neuron in the current layer, and each column corresponds to the connection weights between neurons in the current layer and neurons in the previous layer $h_{(l-1)}$. These weights are learned during the training of the network. $h_{(l-1)}$ the activations from the previous layer $l-1$, which could be the output of a convolutional layer, a pooling layer, or another fully connected layer. It is a vector containing the activations of the neurons in the previous layer.

[35] The operation $W_l * h_{(l-1)}$ involves matrix multiplication between the weight matrix W_l and the activation vector $h_{(l-1)}$. This results in a new vector z_l that contains the weighted sum of the activations from the previous layer, which is then typically passed through an activation function to introduce non-linearity. Common activation functions include the rectified linear unit (ReLU), sigmoid, or hyperbolic tangent (tanh). This fully connected layer operation computes a linear combination of the activations from the previous layer using learned weights, and it's an essential part of the feedforward process in a CNN. The output z_l is then used as the input for subsequent layers or as the final output of the network, depending on the network's architecture and task.

[35] $RELU(z_i) = max(0, z_i)$ represents the operation of the Rectified Linear Unit (ReLU) activation function applied to a single activation z_l in CNN. $RELU(z_l)$ represents the output of the ReLU activation function applied to a single activation value z_l. ReLU is a popular activation function in deep learning, known for its simplicity and effectiveness in introducing non-linearity to the network. z_l is the input activation value to which the ReLU function is applied. This activation value could be the result of a neuron's weighted sum of inputs, computed in a previous layer. $max(0, z_i)$, this ReLU activation function is defined as the maximum of 0 and z_i. In other words, if z_i is positive or zero, it remains unchanged ($RELU(z_i) = z_i$). However, if z_i is negative, it is replaced with 0 ($RELU(z_i) = 0$). This step puts on non-linearity into the neural network, allowing it to learn complex patterns and make the network more expressive.

The ReLU activation function is particularly useful because it mitigates the vanishing gradient problem and accelerates the training of deep neural networks. It has become a standard choice for activation functions in many CNN architectures and deep-learning models. In the context of a CNN, this ReLU activation function is typically applied element-wise to each activation value in a feature map or a fully connected layer. It helps the network learn and model complex relationships in the data, making it capable of recognizing intricate patterns and features during the training process.

$softmax(z_i) = \frac{e^{(z_i)}}{\sum_j^n e^{(z_j)}}$ is used in Convolutional Neural Networks (CNNs) and other neural networks for multi-class classification tasks. It's a function that converts a vector of real numbers into a probability distribution over multiple classes. $softmax(z_i)$ represents the output of the softmax function applied to a single element z_i In a multi-class classification problem, each z_i typically corresponds to the unnormalized logit (score) for one of the classes. e^{z_i} This term computes the exponential of the i-th element z_i in the input vector z. This term essentially measures the "strength" or "confidence" of the model's prediction for the i-th class.$\sum_j^n e^{(z_j)}$ computes the sum of the exponentials of all elements in the input vector z. It represents the total "strength" across all possible classes.

The softmax function takes the exponential of each element z_i and divides it by the sum of the exponentials of all elements in the vector z. This operation scales the elements in z to a probability distribution where each $softmax(z_i)$ represents the probability of the input belonging to the i-th class.

[14] Key properties of the softmax operation include ensuring that the output probabilities sum to 1, making it a valid probability distribution. It amplifies the differences between the elements in the input vector, emphasizing the class with the highest score while reducing the impact of low-scoring classes. It's a differentiable function, which is crucial for training neural networks using gradient-based optimization algorithms like backpropagation. In the context of a CNN for image classification, the softmax layer typically follows the final fully connected layer of the network, and it's used to produce class probabilities. The class with the highest probability is often chosen as the predicted class label for a given input.

CNN has proven successful in extracting local features and using them to predict global features due to its three-layer structure consisting of pooling layers, convolutional layers and fully linked layers. For instance, if CNN's base layer can recognize edges, its advanced layers should be able to recognize more complex objects, such as human faces and parts of the body. CNN's stability and automation are also big factors in why we chose to include it in our model. It is possible for translation or modification to occur while developing a model with deep representation. Movement blurring and noise are particularly problematic with ultrasound imaging. As well as being able to automatically carry out data-driven object representation, CNN has a high tolerance for such issues.

5.1.2 Inception V3

Inception V3 is a deep (CNN) convolutional neural network architecture that was created by Google researchers and is frequently used for classifying medical images. It is a strong and effective model for image recognition because of its innovative architectural design, factorization strategies, and regularization techniques. It is a well-liked option for a pre-trained model because of its effectiveness, outstanding performance, pre-trained weights, and capacity to handle a variety of visual jobs.[2]

It belongs to the Inception family of models, which are made to effectively manage complex image recognition jobs with many parameters. The concepts established in Inception V1 and Inception V2 are expanded upon in Inception V3, which also adds new architectural elements to improve performance.

The Inception V3 pre-trained model has the fundamental characteristics and features including architecture, inception modules, factorization, batch normalization, Global Average Pooling, Regularization of Weight Decay and Pre-trained on ImageNet. Inception V3 architecture layers have 42 layers total, including convolutional, max pooling, average pooling layer, fully connected layers, softmax Layer and dropout layer. It has a deep and detailed structure that makes it possible to extract hierarchical characteristics and sophisticated patterns from photos.

Inception V3 has 42 layers total, including convolutional, pooling, and fully connected layers. It has a deep and detailed structure that makes it possible to extract hierarchical characteristics and sophisticated patterns from photos.

The inception modules of Inception V3 are well-known; these thoughtfully created building blocks enable the network to learn and combine characteristics at various scales and complexities. To capture features of diverse receptive fields, these modules employ a variety of filter sizes (1x1, 3x3, and 5x5). Factorized convolutions are used in Inception V3 to lessen computational difficulty and model size. These are 3x3 and 1x1 convolutional combinations that are computationally more effective than utilizing just 3x3 convolutions alone. Inception V3 includes batch normalization to speed up and stabilize the training process. Throughout training, it adjusts the activations inside each mini-batch, optimizing convergence and generalization.

Inception V3 uses global average pooling rather than completely connected layers at the very end. By averaging the characteristic maps' spatial dimensions, this procedure creates a vector that may be utilized for classification right away. Global average pooling lessens overfitting and results in a model that is more manageable. It uses weight decay regularization to penalize large weights during training, preventing overfitting and improving generalization on unseen data.

This model is pre-trained on the enormous ImageNet dataset, which comprises millions of images that are labeled from a wide variety of categories, just like its predecessors. The pre-training on ImageNet allows the model to learn general traits that, with refinement, may be applied to a variety of additional image identification tasks.

The foundational component of the Inception V3 network is the Convolution layer. It is made up of several filters (little windows) that move over the input image (or featured mappings) and multiply and sum elements in order to detect various features. Each filter picks up on particular patterns within the incoming data, like edges or patterns. [30] The Max Pooling layer is implemented to reduce the dimensions of space of the mappings of features while keeping the most vital data. It separates the input feature map into non-overlapping parts and saves just the highest value from each zone. This procedure aids in the reduction of computing complexity and the prevention of overfitting. The Average Pooling layer, like Max Pooling, does downsizing, but instead of taking the highest possible value from each zone, it calculates the mean of the data. It can assist in smoothing out information and lessen sensitivities to spatial translations in specific systems.

The Inception module of the Inception V3 network employs the Concatenation layer. It merges the outputs of numerous separate filters (concurrent routes) acquired by various Convolution, Max Pooling, and Average Pooling layers into one, bigger feature map. This enables the model to accurately record multi-scale information. The Fully Connected layer, often known as a dense layer, is the typical neural network layer. Every neuron from the preceding layer is linked to every single neuron in the present layer. These fully linked layers are often utilized near the final point of the network in Inception V3 to map the retrieved features to the categories of interest for classification. The Softmax layer is the network's final layer for multi-class classification. The preceding fully connected layer's output is used, and the softmax algorithm is used to convert initial scores (logits) into standardized probability values for each class. Each output node represents a different class, with the node with the highest likelihood representing the anticipated class. The Dropout layer is a regularization technique intended to prevent overfitting during training. During each forward pass, it randomly sets a percentage of the neurons' outputs to zero. This drives the network to hinge on diverse routes and inhibits individual neurons from turning highly specialized on specific data patterns. Throughout testing, the dropout layer is typically disabled. [1]

In general, Inception V3 extracts features that reduce spatial dimensions, combines multi-scale information, and makes class predictions using an assortment of convolutional layers, pooling layers, concatenation, fully connected layers, and the softmax

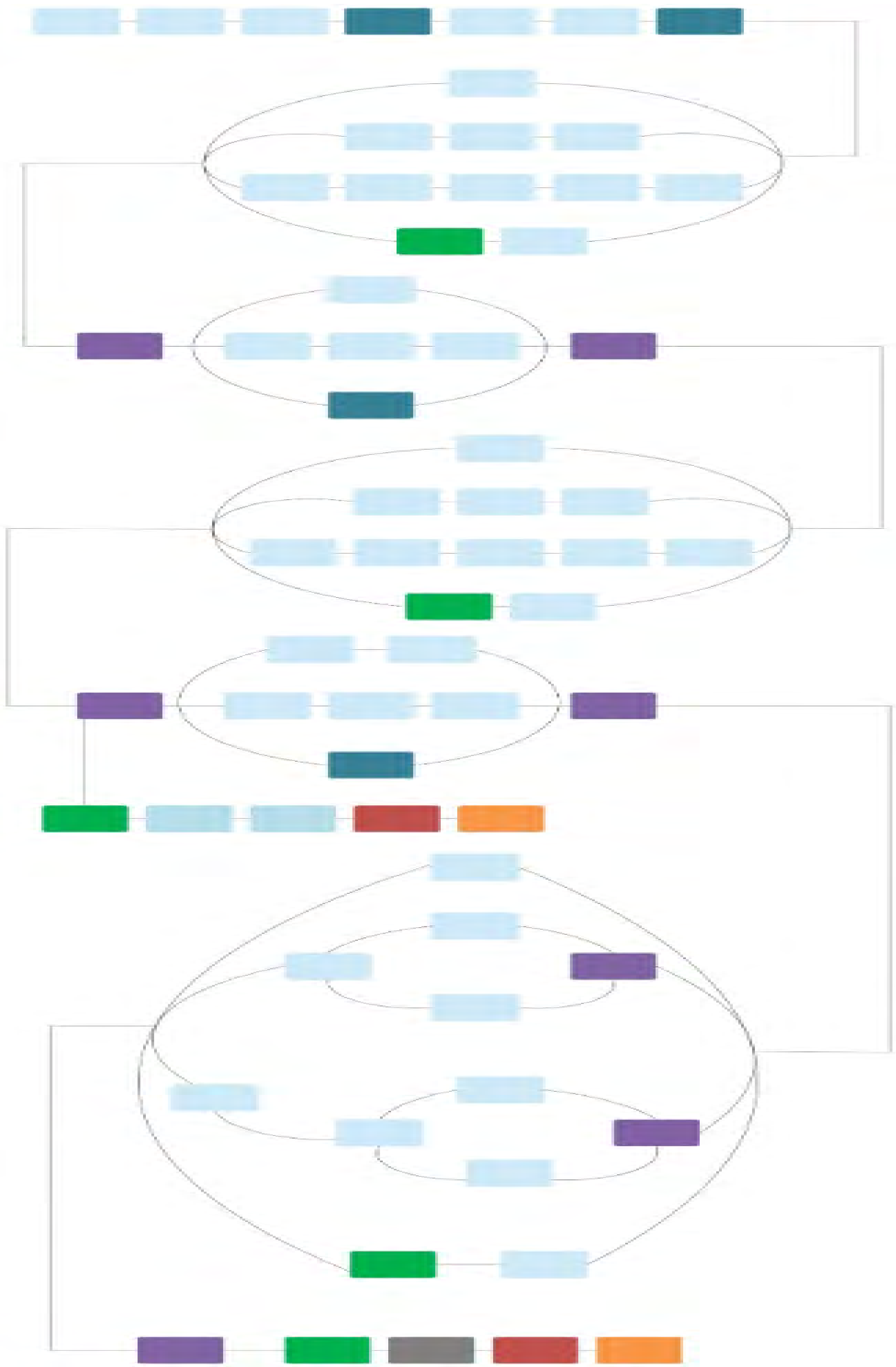


Figure 5.1: Workflow of the layers of Inception V3

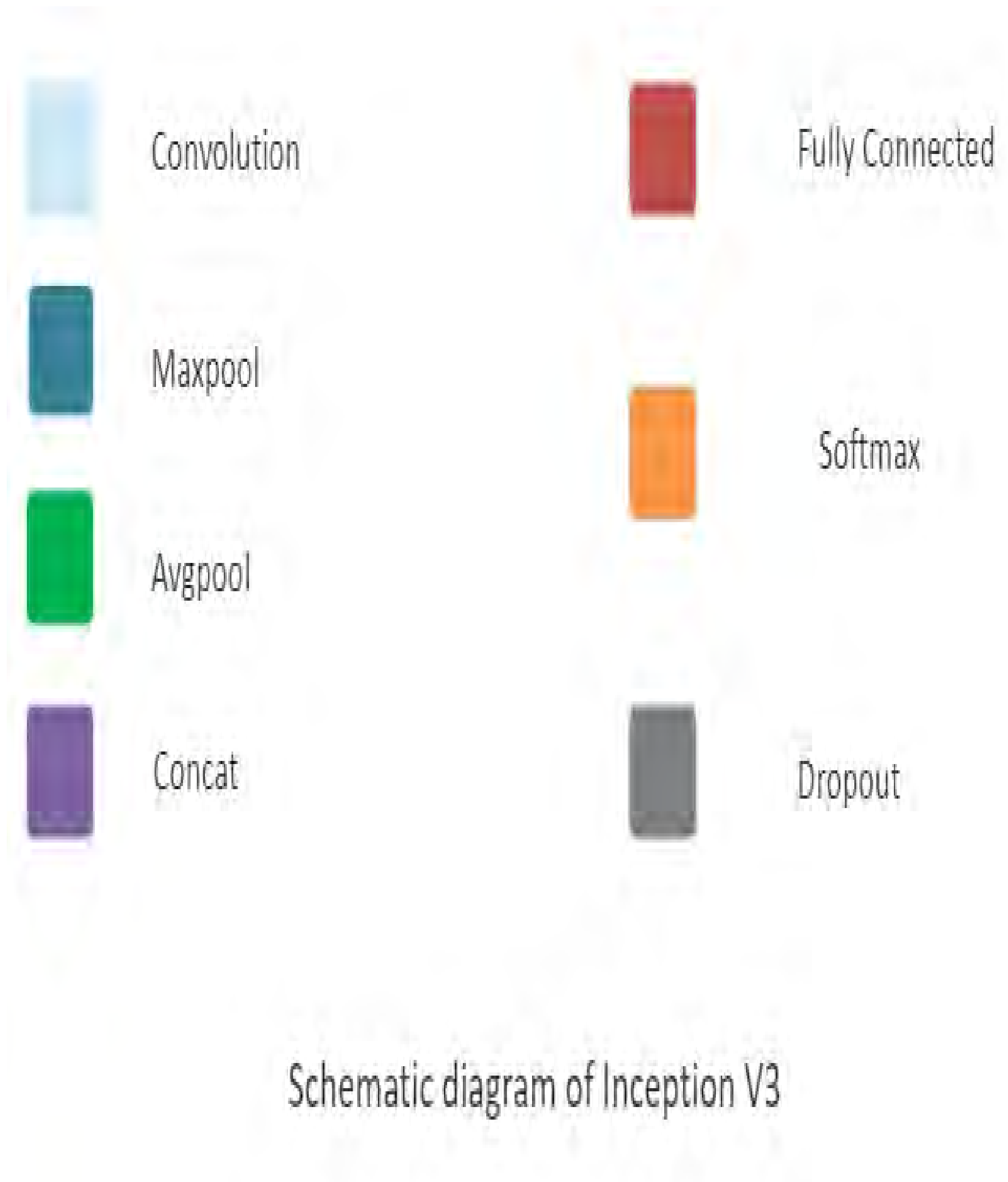


Figure 5.2: Symbols of Inception V3 Schematic Diagram

output layer. The dropout layer is used for improving adaptation during training.

5.1.3 ResNet50

The deep convolutional neural network (CNN) architecture known as ResNet50, or "Residual Network with 50 layers," was developed by Microsoft Research experts. It is a member of the ResNet family that uses residual blocks to tackle the issue of disappearing gradients in very deep neural networks. It consists of Architecture, Residual Blocks, Deep Stacking, Pre-trained on Large Datasets, Transfer Learning, and Efficiency of Computation. [25]

A deep neural network called ResNet50 has 50 layers. Convolutional, pooling and fully linked layers are all combined in this method. The architecture is made to learn increasingly complicated hierarchical features from forthcoming images. The usage of residual blocks is ResNet50's main innovation. By introducing a shortcut connection or skip connection, a residual block enables the model to learn residual mappings or the difference between a layer's output and input. By addressing the vanishing gradient issue, this architecture facilitates the training of extremely deep networks.

[25] ResNet50's depth enables it to learn complex features from photos, capturing both high-level conceptual information and low-level minutiae. This ability is useful for spotting minute structures and patterns in photographs of corneal ulcers. Large picture datasets like ImageNet, which has millions of annotated images over thousands of classes, are frequently used to pre-train ResNet50. With the help of this pre-training, the model is equipped with knowledge of general picture properties that may be applied to and refined for particular tasks, such as the classification of corneal ulcers. It is possible to take advantage of transfer learning by utilizing ResNet50 as a pre-trained model. Transfer learning enables you to use the information gained from one target domain (corneal ulcer images) and apply it to another source domain (ImageNet). The model's parameters can be fine-tuned using photos of corneal ulcers to better suit the specific classification task. ResNet50 is intended to be computationally effective despite its depth, making it ideal for deployment on a variety of hardware and software systems.

The deep learning strategy in the thesis may take advantage of the model's capacity to acquire complicated features, its pre-trained knowledge from ImageNet, and the effective training process made possible by residual connections by using ResNet50 as a pre-trained model for corneal ulcer classification. This combination may result in an automated classification of corneal ulcers that is accurate and trustworthy, thereby benefiting medical professionals in making diagnoses and treatment choices.

WorkFlow of ResNet-50:

ResNet50 is a potent architecture of 50 layers, 50 of which are fully linked, convolutional, and pooling layers. It was created to effectively train incredibly deep networks while addressing the vanishing gradient problem. The workflow of its layers such as input layers, convolutional layers, the layers from 2nd to 50th, global average pooling layer, fully connected layers of 51st to 53rd and softmax activation layer can be summed up as follows when using ResNet50 as a pre-trained model.

The photos of the corneal ulcer are input into the ResNet50 input layer. Usually, the photos are downsized to a predetermined size that is appropriate for the model's input parameters. A number of residual blocks make up ResNet50's central structure. It consists of two or more convolutional layers in each residue block. Batch normalization with ReLU activation comes after the first layer (Layer 2), a 7x7 convolutional layer with 64 filters and a cadence of two. The initial Max Pooling layer then uses 3x3 max pooling with a cadence of 2 to reduce the dimension of the space of the feature maps by a factor of 2. Four groups made up of the remaining 48 layers each comprise some number of residual blocks with various numbers of filters. Three 3x3 convolutional layers with batch normalization and ReLU activation are commonly present in each residual block, and the final layer's filters have a similar number as the input filters. The skip connection (identity shortcut) solves the disappearing gradient issue by connecting a block's input and output together immediately.[20]

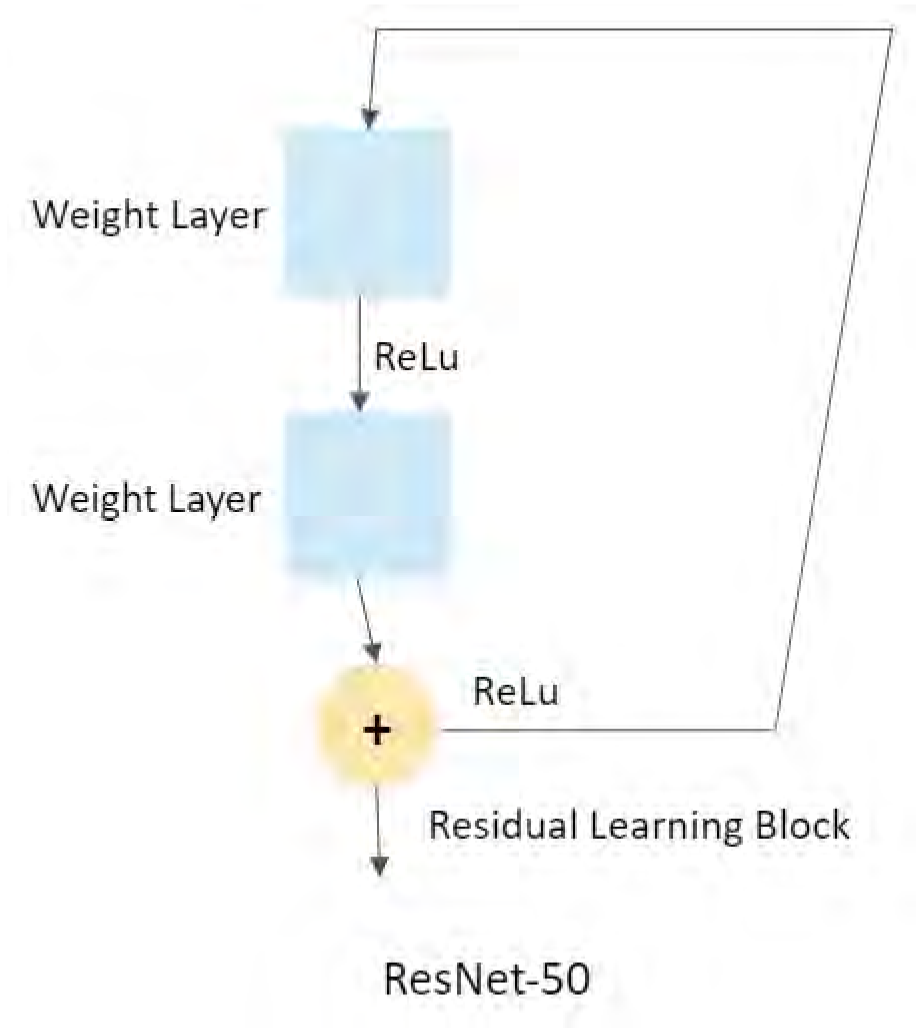


Figure 5.3: Fully Connected Layers of ResNet50

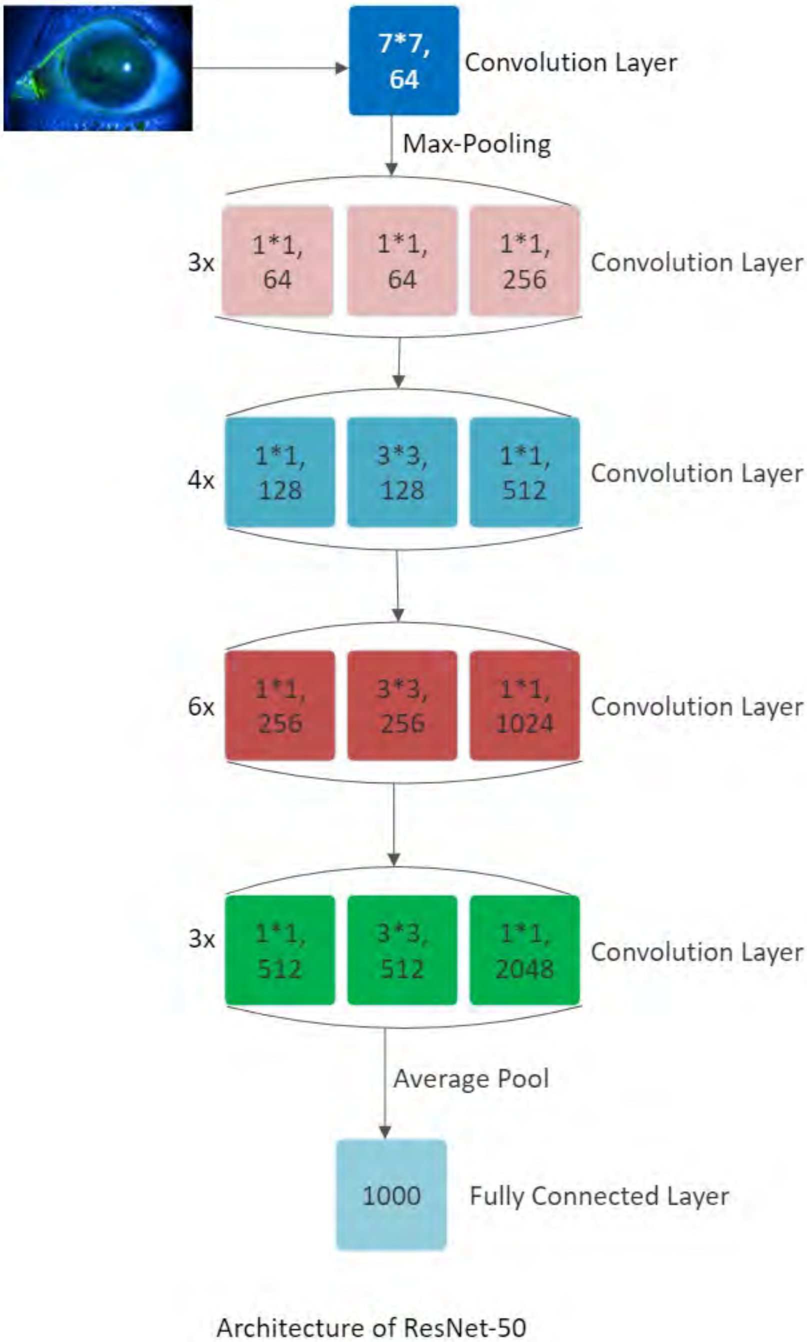


Figure 5.4: Architecture of ResNet50

A Global Average Pooling layer (GAP) is added after the last convolutional layer. The spatial dimensions of every characteristic map are condensed by GAP to one value. It creates the vector format of the image by averaging all the values in every characteristic map. Three fully connected layers (Layer 51, Layer 52, and Layer 53) are linked to the output of the GAP layer. [20] One thousand neurons, or one thousand classes, make up the final fully connected layer of the underlying ImageNet dataset.

The final softmax activation layer turns the probability from the raw scores of the previous fully linked layer into. The possibility that an input image belongs to a given class is indicated by each class score. The final fully connected layer with 1000 classes (from ImageNet) is typically replaced when using ResNet50 for the automated classification of corneal ulcers with the appropriate number of classes such as a binary classification for "corneal ulcer" vs. "non-corneal ulcer".

To sum up, the ResNet50 layer procedure entails a number of convolutional layers, residual blocks, global average pooling, and fully connected layers, along with a softmax activation layer serving as the final layer that generates likelihood for various classes. ResNet50 may be fine-tuned to classify corneal ulcers more accurately and quickly by using its pre-trained knowledge.

5.1.4 VGG16

[25] VGG-16 is a deep convolutional neural network (CNN) architecture , also called the Visual Geometry Group model 16, which was created by researchers at the University of Oxford. It is distinguished by the straightforwardness and widespread application of 3x3 convolutional filters. In order to investigate and comprehend the effects of network depth on image recognition tasks, it was included as a component of the VGG project. One of the variants of the VGG family is called VGG16, and the number "16" in its name denotes how many layers it has. The VGG16 pre-trained model's salient qualities and traits include Architecture, Standardized filter size, Dimension and Stacking, Spatial Dimensions Reducing, Activation Function, Fully Connected Layers, Pre-trained on ImageNet, Popularity and Learning Transfer.

The 16-layer VGG16 deep neural network has 13 convolutional layers, 3 fully connected levels, and other layers. A succession of tiny 3x3 convolutional filters, followed by max-pooling layers, make up the architecture, which is quite straightforward and results in a fairly uniform and regular structure. VGG16 uses only 3x3 filters throughout the network, unlike some other CNN architectures (such as Inception modules), which make use of filters of various sizes. The architecture is made simpler to train and interpret the constant filter size.

The VGG16 design is more complicated than many earlier CNN architectures, and because of its stacked layers, the model may learn hierarchical characteristics that are more complex. Deeper networks have greater feature extraction capabilities because they can capture more abstract representations. To minimize the spatial dimensions of feature maps, VGG16 employs max-pooling layers with 2x2 filters and a stride of 2. This makes the network more effective by downsampling the feature maps while keeping key information. The Rectified Linear Unit (ReLU) activation function, which is used in VGG16, makes the model non-linear. ReLU activation speeds up training by avoiding vanishing gradients and enables the model to learn complex patterns.[11]

VGG16's final layers are completely connected layers that result in the classification of the input image into various groups. These layers integrate the previously learned features and generate likelihoods of classes. VGG16 was trained using the ImageNet dataset, which comprises millions of annotated images from various categories. Even with insufficient training data, the prior training on ImageNet allows the model to acquire general characteristics that can be fine-tuned for diverse picture identification tasks. VGG16 has grown in popularity due to its ease of use and strong performance on image recognition tests. It has become the industry standard for analyzing the efficiency of various CNN systems. Its pre-trained weights are also commonly employed for transfer learning in a variety of applications involving computer vision.

It is a well-liked option for a variety of computer vision tasks, such as image categorization, object identification, and extraction of features, because of its simple design and weights that have been pre-trained. It continues to be a pertinent and significant model in the sector of computer vision and offers a crucial foundation for deep learning researchers as well as practitioners.

The layers' operation of VGG-16 can be stated up as follows: The photos of the corneal ulcer are entered into the VGG-16 input layer. Typically, the photos are downsized to a set size (such as 224x224 pixels) that is appropriate for the model's input specifications.

The number of stacks of convolutional layers (1-13) provide the foundation of the VGG-16 architecture, which is then topped with max-pooling layers. The configuration of each convolutional layer is 3x3 filters with a stride of 1. Batch normalization and a ReLU activation function are applied after the first convolutional layer (Layer 1). Up to Layer 13, a similar structure is applied to the following convolutional layers. [15]

In the max pooling layers 14, 17 and 20, the VGG-16 applies it with a 2x2 filter and a stride of 2 for each two convolutional layers. Downsizing is made easier by the max-pooling process, lowering the spatial dimensions of the feature maps by a factor of 2.

VGG-16 has 21, 22, 23 three completely linked or fully connected layers following the stack of convolutional and pooling layers. 4,096 neurons make up Layer 21, the first completely linked layer, and are followed by batch normalization and a ReLU activation function. There are 4,096 neurons in the second fully linked layer (Layer 22), which is followed by batch normalization and a ReLU activation function. The number of neurons in the final fully linked layer (Layer 23) is equal to the number of output classes.

The final softmax activation layer turns the probability from the raw scores of the previous fully linked layer into the probability that an input image belonging to a specific class is represented by each class value.

It's vital to remember that the VGG-16 model was initially trained using the 1,000-class ImageNet dataset. A new fully connected layer having the necessary number of neurons for the particular classification job will be used in place of the final fully connected layer's 1,000 neurons (equivalent to ImageNet classes) for the categorization of corneal ulcers. Convolutional layer stacking and max-pooling layers are used in VGG-16's layer workflow to extract hierarchical features, which are then followed by fully connected layers to generate class predictions. [26] The VGG-16 model's capacity to acquire complicated characteristics and produce precise classification outcomes can be utilized by the deep learning approach by employing it as a model that has previously been trained and adjusting it on corneal ulcer images.

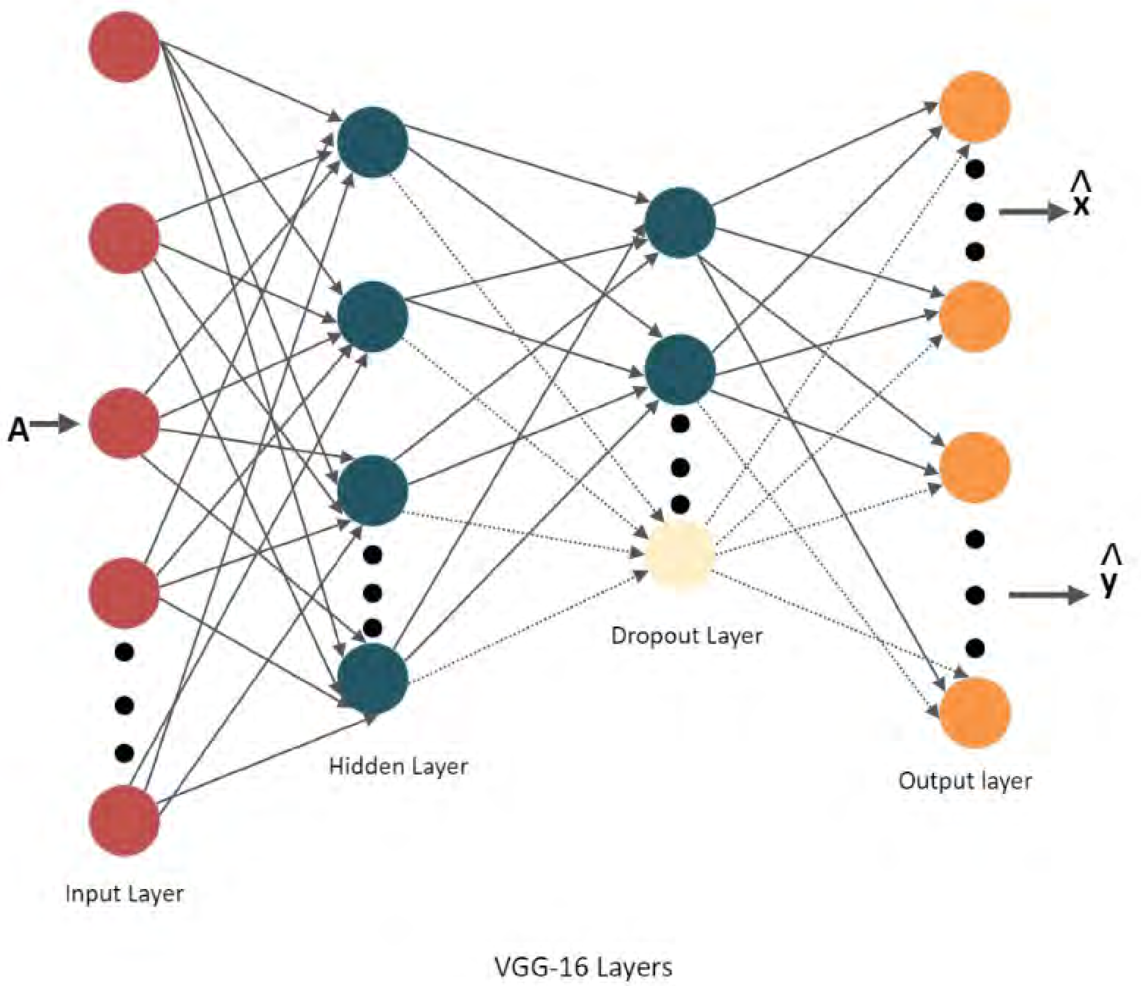


Figure 5.5: VGG16

Figure 5.6: Operation Layers of VGG16

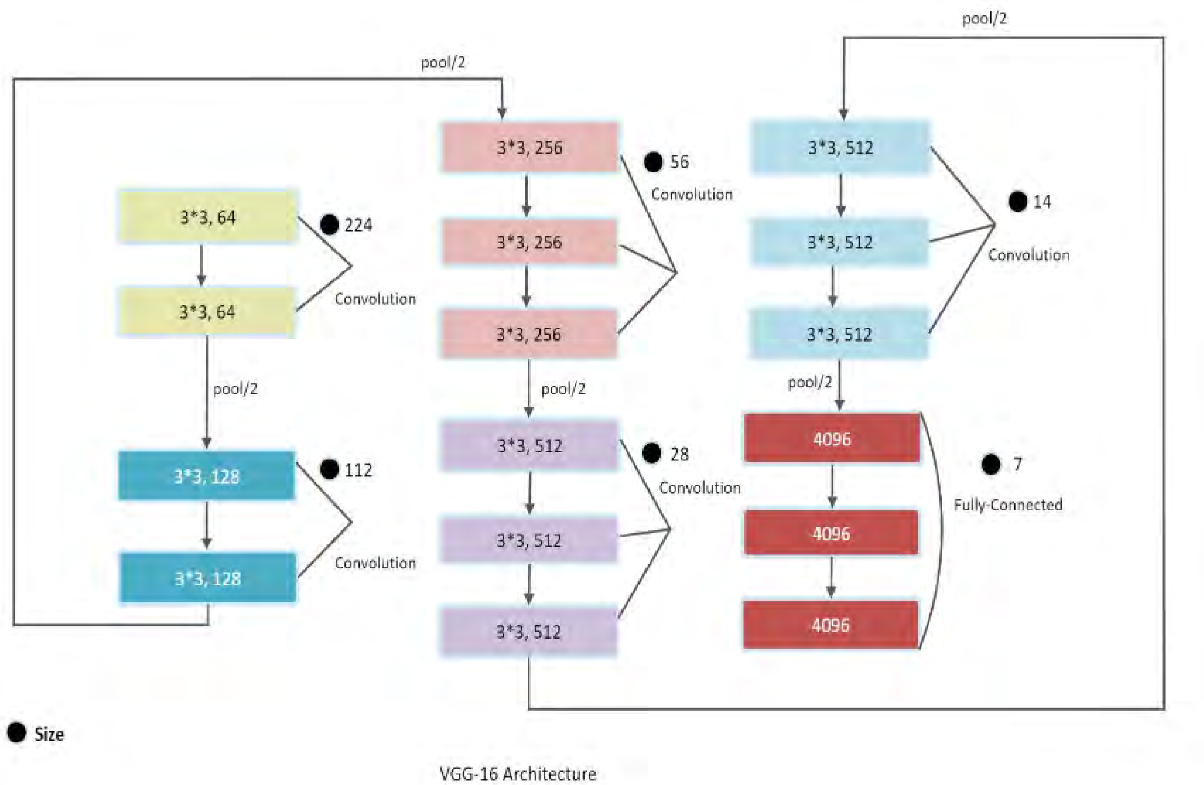


Figure 5.7: Architecture of VGG16

5.2 Customized Model

A customized model, also known as a fine-tuned model, is a neural network that has undergone specific modifications or adaptations to cater to a certain task or domain. The process starts with an initial model that has been pre-trained on a comprehensive dataset, enabling it to acquire general features. Subsequently, this model undergoes further training on a smaller dataset that is tailored to the job at hand. The aim is to use the knowledge included inside the pre-trained model while customizing it to perform exceptionally well in a particular task, such as image categorization, object recognition, or natural language understanding.

[9] The present research used a Convolutional Neural Network (CNN) architecture to develop a tailored model. CNN (Convolutional Neural Network) involves several components in the construction of a customized model. To begin, it is necessary to choose a pre-trained convolutional neural network (CNN) model that is suitable for the dataset and tasks at hand. Pre-existing models such as VGG, ResNet, Inception, MobileNet, and others are often used in Deep Learning frameworks like TensorFlow, Keras, and PyTorch. Furthermore, due to the specific nature of the work and its lack of applicability to customized jobs, it is recommended to remove the uppermost layers, which are fully connected layers, from the pre-trained model. Additionally, the newly implemented layers that meet the requirements of the task output criteria have the potential to be substituted. Furthermore, it is recommended to include additional layers into the model that are specifically designed for the given purpose. Examples of these layers include fully connected layers, additional convolutional layers, and activation functions. Next, it is necessary to provide a definition for the loss function, optimizer, and metrics used in the customized model. The choice of these elements is contingent upon the particular objective, such as categorization, prediction, and the nature of the dataset. Next, it is important to create datasets that are appropriate to the work at hand. This involves many steps such as importing the data, preparing it by doing tasks like image scaling, normalizing pixel values, and maybe using data augmentation techniques if deemed essential. In addition, it is recommended to use a dataset that is specifically designed for the job at hand in order to train the customized model. In practice, transfer learning is often used to refine the model by using pre-existing knowledge from image data. The training process should be monitored in order to identify any essential modifications that need to be made to the hyperparameters. [8] Subsequently, the evaluation of the model's accuracy and other essential metrics may be conducted by executing it on an alternative validation dataset. The last step involves doing more fine-tuning of the model via the exploration of hyperparameters, model architecture, and regularization techniques. In order to get the required level of performance, it is necessary to optimize the model. Ultimately, when the customized model has undergone training and refinement, it may be used to create predictions or classifications on new, unexplored data.

In the domain of model training, a diverse range of terminologies are used. The model utilizes TensorFlow and Keras, which are integrated libraries often used for the construction and modification of deep learning models. The definitions of the terms are expounded upon in further depth below.

1. **Tensorflow:** TensorFlow is an open-source framework for machine learning and deep learning developed by Google. TensorFlow was used to develop and train customized models, namely neural network models, by using its adaptable nature and advanced machine learning capabilities. [7] The framework provides a comprehensive and resilient structure for developing a diverse array of deep learning and machine learning applications using photographic data.

   The TensorFlow API hierarchy consists of higher-level APIs that are built upon lower-level APIs. Low-level application programming interfaces (APIs) are used by researchers in the field of machine learning for the purpose of developing and evaluating novel machine learning algorithms. During this workshop, participants will engage in the process of creating, training, and forecasting machine learning models using the tf.keras high-level API. The tf.keras library serves as the TensorFlow implementation of the Keras API, which is a widely-used open-source framework. The graphic below illustrates the hierarchical structure of the TensorFlow toolbox.

2. **Relu:** The Rectified Linear Unit (ReLU) is an activation function that is often used in neural networks and other models within the field of deep learning. A deep learning model that incorporates non-linearity has the ability to acquire intricate patterns and representations inside datasets. The activation function used by the Rectified Linear Unit (ReLU) is defined as

   f(x) = max(0,x).

   In this context, the variable x is used to denote the input function, whereas f(x) is used to indicate the output function. In the case when x is more than or equal to zero, the maximum function will yield the value of x. Conversely, if x is less than zero, the maximum function will provide a value of zero.

   The inclusion of Relu is integral to the comprehensive coverage provided by CNN. The use of the Rectified Linear Unit (ReLU) is employed to generate activation values, hence facilitating nonlinearity within the model. This operation involves the selection of a certain group of values, which are then assigned a value of zero. The proposed operation involves taking a feature map and substituting any negative values (-1) with zero (0). In the event that the value exceeds zero, it will stay unaltered. Moreover, it enhances the speed of both training and calculation processes.

3. **Conv2D:** The Conv2D, also known as the 2D Convolutional Layer, is a fundamental component of convolutional neural networks (CNNs), which are often used in the field of deep learning for tasks involving images. Convolutional techniques are often used to process input data, typically consisting of images or feature maps, with the objective of acquiring knowledge and extracting hierarchical features.

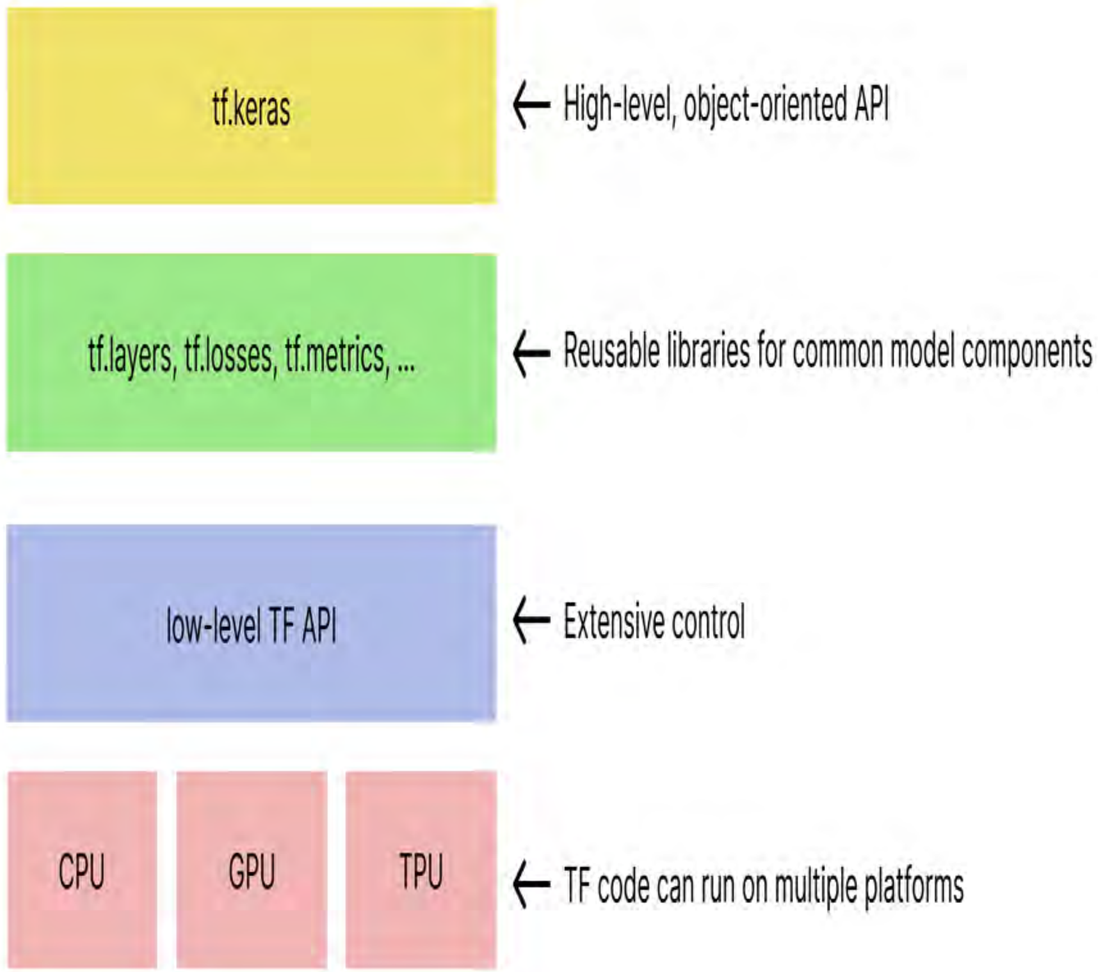


Figure 5.8: Hierarchical Structure of Tensorflow

4. **Flatten:** The "Flatten" layer, referred to as Flatten(), is a kind of layer often used in neural networks, particularly in deep learning architectures like convolutional neural networks (CNNs) and fully connected neural networks (FCNNs). The primary purpose of the flattened layer is to reconfigure the input data, which is initially in the form of a multi-dimensional tensor with spatial dimensions, into a one-dimensional tensor, also known as a vector. The transition from convolutional layers to fully linked layers necessitates this modification in a neural network.

   The input of the flattened layer typically consists of a multidimensional tensor, which is often obtained from a previous layer, such as a Conv2D layer in a convolutional neural network (CNN). The flattened layer transforms the input tensor into a unidimensional vector by sequentially arranging all the values, hence eliminating any spatial arrangement. The use of the flattened layer is common when transitioning from the convolutional component of a neural network, responsible for extracting spatial features from input data such as images, to the fully connected part of the network that processes the flattened feature vector. The inclusion of a flattened layer is crucial in neural networks since it enables the use of fully connected layers, which need one-dimensional input. The transition from convolutional layers, which process spatial input, to fully linked layers is facilitated.

5. **Dense Layers:** In the realm of neural network architectures, dense layers, sometimes referred to as fully connected layers or simply "Dense" layers, have significant importance as fundamental elements inside deep learning models. The inclusion of these layers in various neural network topologies is of utmost importance due to their role in the interpretation and acquisition of intricate patterns and representations in the input data.

   There are no limitations or constraints imposed on the use of thick layers within Convolutional Neural Network (CNN) architectures. Nevertheless, it is essential that the ultimate dense layer include an appropriate number of neurons and activation function, with an optimal representation of the task at hand. The implementation of layers with two thick levels must adhere to the following guidelines.

   In the context of neural networks, a Dense layer receives a one-dimensional tensor, sometimes known as a vector, as its input. This tensor represents either the output of the preceding layer or some processed data. The output of a Dense layer is a one-dimensional tensor, often referred to as a vector. Each element inside this vector corresponds to the activation value of an individual neuron within the layer. Every entry in the output vector corresponds to a neuron or unit inside the Dense layer. The neurons in question are connected to each element of the input vector. Each connection inside the Dense layer, linking an input element to a neuron, has an associated weight. In the process of constructing a neural network, the quantity of neurons inside a Dense layer is a hyperparameter that may be modified. The selection of the number of neurons used is contingent upon the specific task at hand and the underlying architecture of the model. Dense layers are often used throughout the "fully connected" segment of neural networks to facilitate the integration and analysis of data obtained from preceding layers. They play a crucial role in a wide range of applications, including image classification, regression analysis, natural language processing, and several other domains.

6. **Hyperparameter:** In the context of convolutional neural networks (CNNs), hyperparameters refer to parameters that are not subject to learning from training data and need explicit specification prior to the commencement of training. The selection of hyperparameters has a crucial role in determining the architecture and functionality of the network. The right tuning of hyperparameters is of utmost importance in order to achieve optimal model performance.

   The hyperparameters exert direct control on the structure, function, and performance of the model. The optimisation of model performance may be achieved by data scientists via the adjustment of hyperparameters. The aforementioned technique plays a crucial role in the achievement of machine learning outcomes, and the selection of ideal hyperparameter values has significant importance.

   The hyperparameter in question determines the quantity of filters or kernels that will be used in every convolutional layer. The determination of the receptive field, which refers to the specific area of the input that is evaluated by each filter, is contingent upon the dimensions of the convolutional filters, often denoted as height by width. The sizes that are most often seen are 3x3, 5x5, and 7x7.

   The stride parameter determines the step size while moving the convolutional filter over the input. Padding may be categorized into two types: "valid" padding, which involves no padding, and "same" padding, which entails zero padding. The 'valid' option results in a decrease in the output dimensions, whereas the 'same' option keeps the same dimensions as the input. The selection of pooling type, whether it be max pooling or average pooling, as well as the choice of pooling window size, play a crucial role in determining the reduction of spatial dimensions inside pooling layers.

   A hyperparameter refers to the activation function used in each layer, such as Rectified Linear Unit (ReLU), sigmoid, or hyperbolic tangent (tanh). Due to its high effectiveness, the Rectified Linear Unit (ReLU) is often used as the default option. The determination of the number of convolutional layers, pooling layers, and fully connected layers in the architecture of a Convolutional Neural Network (CNN) is an important hyperparameter.

Chapter 6

Model Implementation

6.1 Environment setup

The model is put into action in a Python environment that has been set up with the most important deep learning tools. The following things stand out:

We relied on a graphics card manufactured by NVidia; the model number is RTX 3070Ti, and it comes equipped with 8 GB of memory; however, the onboard RAM in our computer system is 32 GB. Windows 11 was the version of the operating system that we used. When we used this configuration to train our dataset, it took around an hour of time. We worked with Cuda version 11.2, and in addition to that, we made use of the cuDNN, which was of immense help to us in terms of various libraries. Python 3.8.17 was installed in the environment of conda along with the Tensorflow version 2.13.0. The code is set up to work with picture data in a certain directory. There is a goal directory and an output directory. In the other directory, the information is separated into groups (grades). The data is read from the given directories, resized to a common resolution, and split into training and testing sets. During this step, techniques for adding more data to the images are used. GPU boost is used to make training go faster. TensorFlow's support for GPUs is used, and GPU memory growth is turned on to make the best use of RAM.

6.2 Structure of The Model:

LAYER (TYPE)	**OUTPUT SHAPE**	**Param**
INPUT$_1$($INPUTLAYER$)	(NONE, 128, 128, 3)	0
$conv2d_1$	(NONE, 128, 128, 32)	896
$max_pooling2d$	(NONE, 64, 64, 32)	0
$conv2d_2$	(NONE, 64, 64, 64)	18496
$conv2d_3$	(NONE, 64, 64, 64)	36928
max$_pooling2d_1$	(NONE, 32, 32, 64)	0
$conv2d_4$	(NONE, 32, 32, 128)	73856
$conv2d_5$	(NONE, 32, 32, 128)	147584
$conv2d_6$	(NONE, 32, 32, 128)	147584
max$_pooling2d_2$	(NONE, 16, 16, 128)	0
$conv2d_7$	(NONE, 16, 16, 256)	295168
$conv2d_8$	(NONE, 16, 16, 256)	590080
$conv2d_9$	(NONE, 16, 16, 256)	590080
max$_pooling2d_3$	(NONE, 8, 8, 256)	0
$conv2d_{10}$	(NONE, 8, 8, 512)	1,180,160
$conv2d_{11}$	(NONE, 8, 8, 512)	2,359,808
$conv2d_{12}$	(NONE, 8, 8, 512)	2,359,808
max$_pooling2d_4$	(NON$E, 4, 4, 512$)	0
$conv2d_{13}$	(NONE, 4, 4, 1024)	4,719,616
$conv2d_{14}$	(NONE, 4, 4, 1024)	9,438,208
max$_pooling2d_5$	(NONE, 2, 2, 1024)	0
$conv2d_{15}$	(NONE, 2, 2, 512)	4,719,104
$conv2d_{16}$	(NONE, 2, 2, 256)	1,179,904
batch$_normalization$	(NONE, 2, 2, 256)	295,040
$conv2d_{17}$	(NONE, 2, 2, 128)	512
max$_pooling2d_6$	(NONE, 1, 1, 128)	0
dropout	(NONE, 1, 1, 128)	0
flatten	(NONE, 128)	0
dense	(NONE, 5)	645
Total params:		28,162,725
Trainable params:		28,162,469
Non-trainable params:		256

The structure of the model is made up of a series of convolutional layers and then a series of max-pooling layers. Of note:

Input Layer: Images with 3 color (RGB) channels that are 128x128 pixels in size can be used as input.

Convolutional Layers: There are seven sets of convolutional layers (Conv1 through Conv7). Each set has a number of convolutional layers with filter sizes that get bigger with each layer. Weights are turned on and set up with Rectified Linear Unit (ReLU) activation functions and He-normal setup.

Max-Pooling Layers: After each set of convolutional layers, max-pooling is used to lower the number of spatial dimensions.

Flattening and Dense Layer: The feature maps are flattened, and then a dense layer with softmax activation is used to get class odds. There are five classes, which represent different grade levels.

Batch Normalisation and Dropout: Batch normalization is used to improve convergence and to make the process of training more stable, and dropout is used to stop overfitting and make the model more accurate.

Model Compilation: The model is put together using category cross-entropy loss and the Adam optimizer with a certain learning rate. Accuracy is used as a measurement.

Callbacks: Callbacks are used for model checkpoints, early stopping, and TensorBoard logging so that the training progress can be seen and saved.

- **ModelCheckpoint:** To save the best model weights.
- **EarlyStopping:** To halt training if validation accuracy does not improve.
- **TensorBoard:** For logging training progress and visualizing metrics.

6.3 Flowchart of the custom model

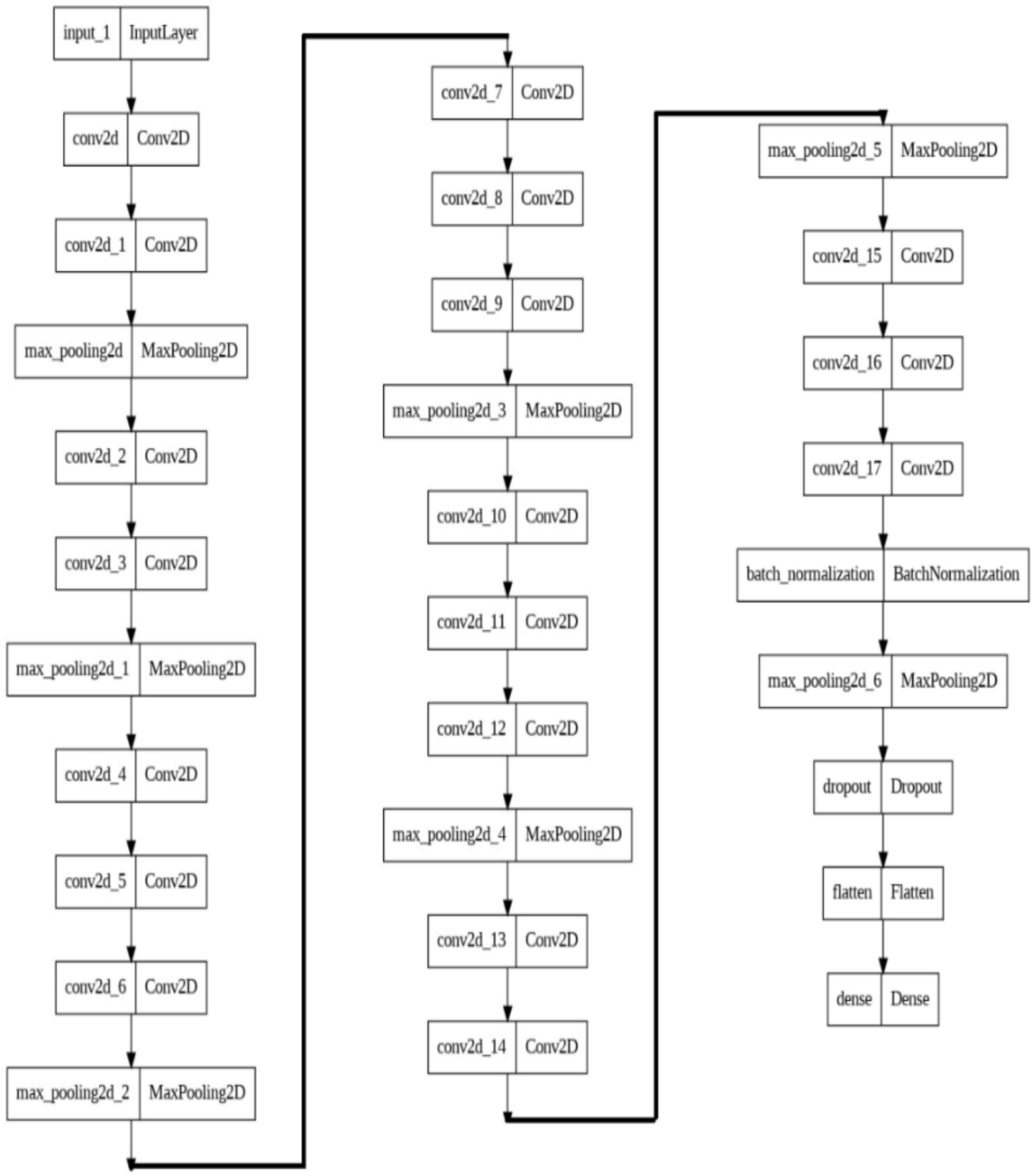


Figure 6.1: Flowchart of the custom model

6.4 Layered view of the custom model

The Visualkeras library offers a straightforward solution for visualizing the layered representation of a customized model in a clear and appealing way. [5] The process of visualizing the architecture of a neural network using visualkeras entails generating a layered diagram that accurately represents the model's hierarchical framework, including its multiple layers and interconnections. This visualization fulfills multiple objectives:

Model Understanding: The graphical representation of the model offers a user-friendly and visually clear means for researchers, and engineers in neural network tasks to quickly understand the fundamental structure.

Debugging and Troubleshooting: Debugging and Troubleshooting are essential processes while developing customized neural network architectures, as it is common to have errors connected to layer connections or dimensions. The utilization of visualization techniques can speed up the identification and solution of these challenges.

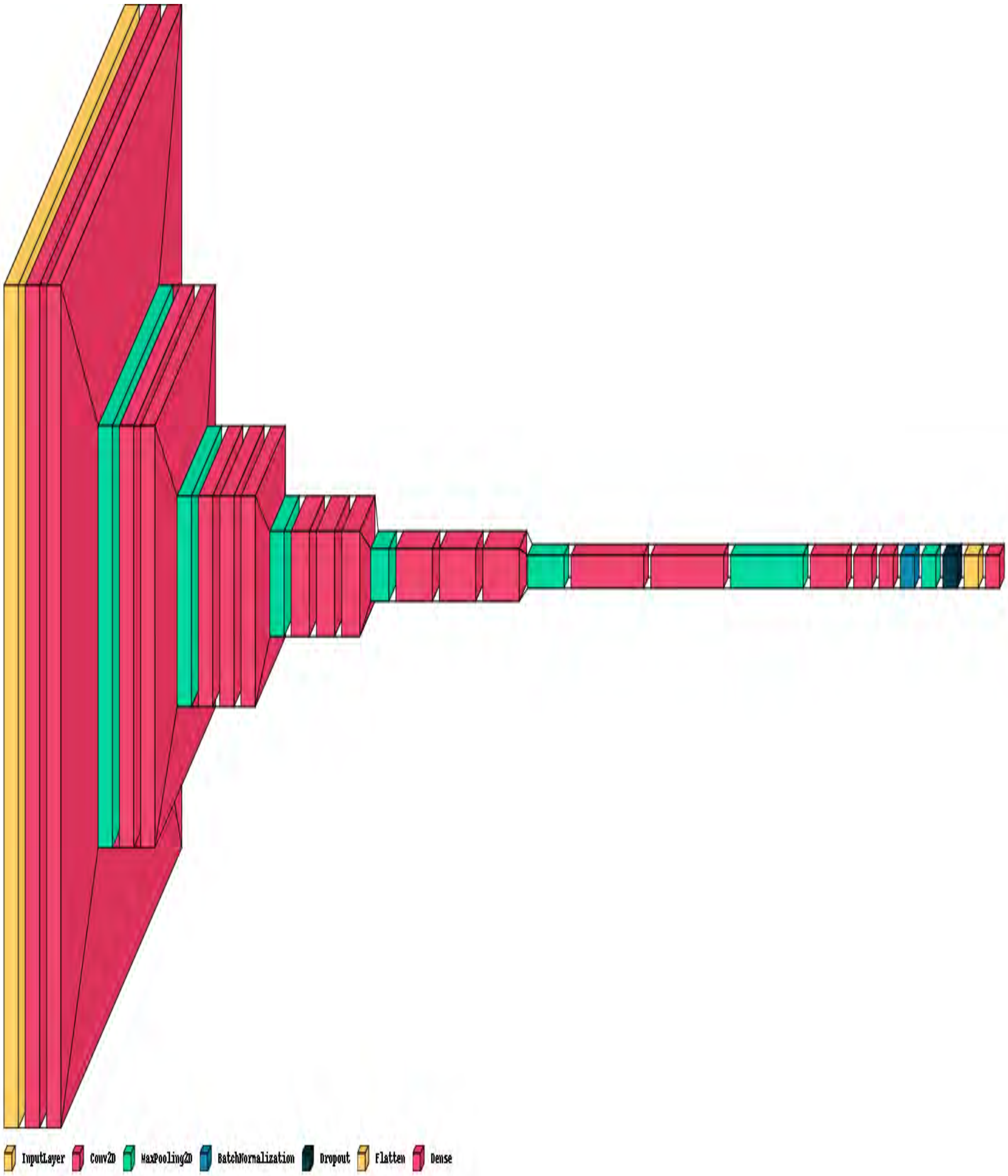


Figure 6.2: Layered view of the custom model

6.5 Transferring Learning with Keras:

The model that was put into place is a custom convolutional neural network (CNN) framework made for classifying images. Some important design decisions are:

Layer Architecture: The model is made up of multiple convolutional layers (Conv1 to Conv7) with rising filter sizes, followed by max-pooling layers. This is a deep architecture that was made to hold traits that are hard to describe.

Transfer Learning: This model doesn't use pre-trained weights from a different job or dataset like traditional transfer learning does. Instead, it starts from scratch and learns how to use the corneal ulcer dataset to find important features.

Data Augmentation: Techniques for adding more data to the pictures are used to make the models more general. These include things like rotating and flipping the data that was put in by chance.

Model Training: The model is trained using category cross-entropy as the loss function and the Adam optimizer with a certain learning rate of 0.0001. Both training data and confirmation data are used to keep an eye on training.

Early stopping: This is done to avoid overfitting. Training stops if the accuracy of validation doesn't get better after a certain number of epochs.

In short, the code sets up an environment for image classification, implements a custom CNN architecture for corneal ulcer grade classification, and uses important deep learning techniques like data augmentation, batch normalization, and dropout for model training and generalization. This method does not use weights that have already been learned, so it is a unique way to apply transfer learning to a specific dataset.[7]

Chapter 7

Result and Model Analysis

7.1 Comparison between all pre-trained model

In this part, a complete comparison is presented between four widely recognized pre-trained deep learning models, namely ResNet50, Inceptionv3, VGG16, and InceptionResNetV2, with our own model. The performance of these models was assessed based on their ability to classify corneal ulcer grades in a collection of medical images. The comparison includes several aspects, including accuracy, loss, and computing efficiency.

Table 7.1: Pre-Train model Validation Result

Model Name	Validation Loss	Validation Accuracy
ResNet50	0.5623	0.85
Inception V3	2.15	0.48
InceptionResNetV2	1.6104	0.2006
VGG16	0.67	0.84

Table 7.2: Pre-Train model Training Result

Model Name	Training Loss	Training Accuracy
ResNet50	0.0038	0.99
Inception V3	0.5	0.8
InceptionResNetV2	1.6046	0.2107
VGG16	0.002	0.99

Table 7.3: Pre-Train Model Precision, Recall, F1 Score Results

Model Name	Precision	Recall	F1 Score
ResNet50	0.8290	0.8297	0.8290
InceptionV3	0.4897	0.4843	0.4822
InceptionResNetV2	0.2621	0.2007	0.0813
VGG16	0.8465	0.8473	0.8468

The ResNet50 architecture, which is widely recognized for its use of residual connections, shows a remarkable accuracy rate of 85% when tested on the validation dataset. This result indicates the system has the ability to accurately identify and analyze important characteristics for the classification of corneal ulcer grades.

The Inceptionv3 model, known for its use of inception modules with a complicated architectural design, demonstrates a validation accuracy of 48%, showing a comparatively lower level of performance. It also means that the model can have difficulties extracting significant characteristics from the data, possibly as a result of complicated architectural complexities. The combination of Inception and ResNet, known as InceptionResNetV2, exhibits a significantly reduced validation accuracy, measuring only 19.9%. This observation implies that the model may have difficulties in performing this specific classification task, possibly because of its higher level of complexity.

The VGG16 approach, renowned for its simple and reliable architecture, obtains a validation accuracy of 82%. Although VGG16 presents a simpler architecture in comparison to ResNet and Inception models, it demonstrates commendable performance in the context of this classification test.

7.2 Result of Customized Model:

Table 7.4: Custom Model Results

Training Accuracy	Training Loss	Validation Accuracy	Validation Loss
0.99	0.063	0.90	0.43

7.2.1 Custom Model Training-Testing Accuracy Result

The performance of the customized model is further evaluated by comparing the accuracy obtained on the validation dataset with the accuracy gained on the training dataset. This comparative analysis helps in determining if the model demonstrates overfitting or underfitting.
The training accuracy of the customized model has an outstanding performance of 99%, indicating a strong ability to successfully identify the patterns within the training data. The observation of high training accuracy suggests that the model holds the ability to effectively understand the complex patterns and features included within the training dataset.
On the other hand, the validation accuracy, although still commendable at 90%, indicates a little decrease compared to the training accuracy. The observed mismatch suggests a minor level of overfitting, wherein the model demonstrates little improvement in performance on the training data compared to unobserved validation data. Nevertheless, the probability of overfitting is rather moderate, although the model's performance continues to prove persistence and efficiency.

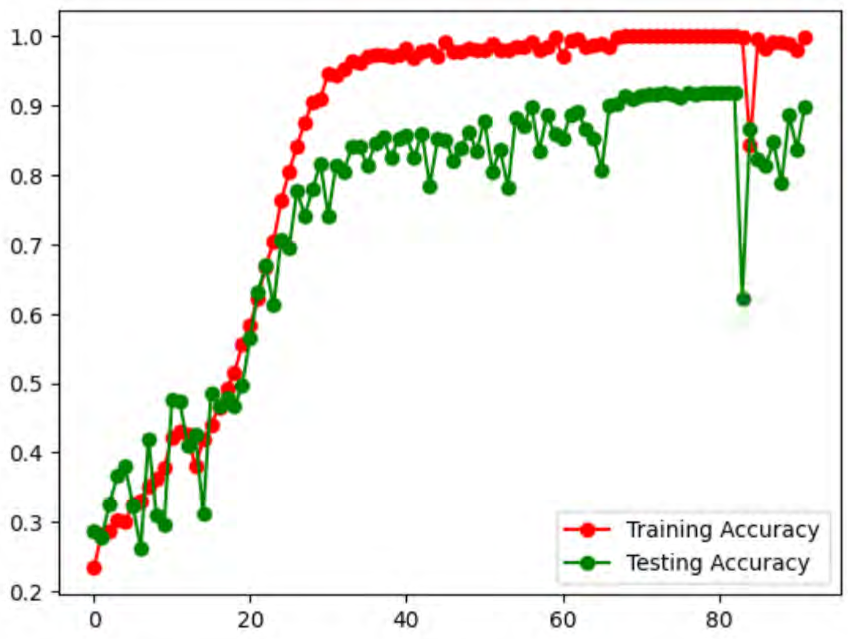


Figure 7.1: Custom model Training-Testing Accuracy Graph

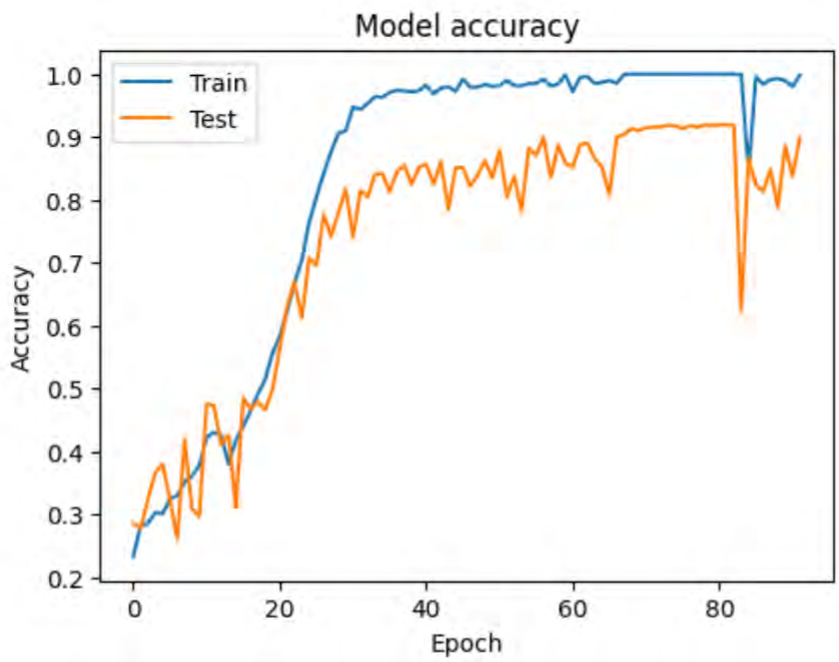


Figure 7.2: Custom Model Train-Test Accuracy vs Epoch Graph

7.2.2 Custom Model Training-Testing Loss Result

A parallel comparison is presented between the validation loss and training loss of the customized model. These metrics offer valuable insights regarding the model's ability for generalization.

The training loss of the customized model is impressively low, measuring at 0.063. This indicates the model's proficiency in efficiently reducing the margin of error on the training dataset. The observation of minimal training loss indicates that the model has effectively captured the fundamental patterns that exist in the training data.

The validation loss, although marginally increased at 0.43, remains within an acceptable range. The result shows that the model has strong generalization capabilities when applied to new, unseen data. Additionally, the difference between the training and validation loss is not excessively great, which further supports the model's robustness and reliability.

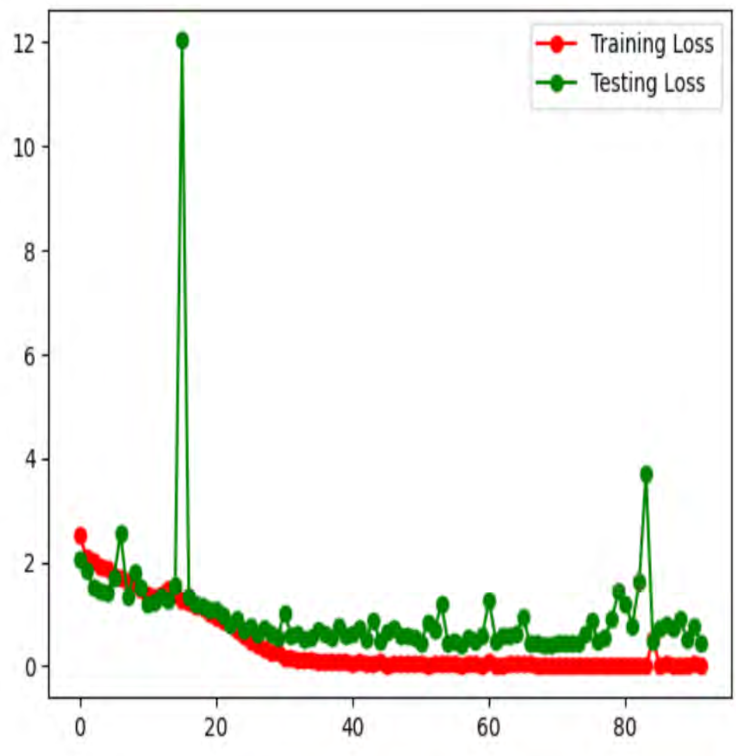


Figure 7.3: Custom model Training-Testing Loss Graph

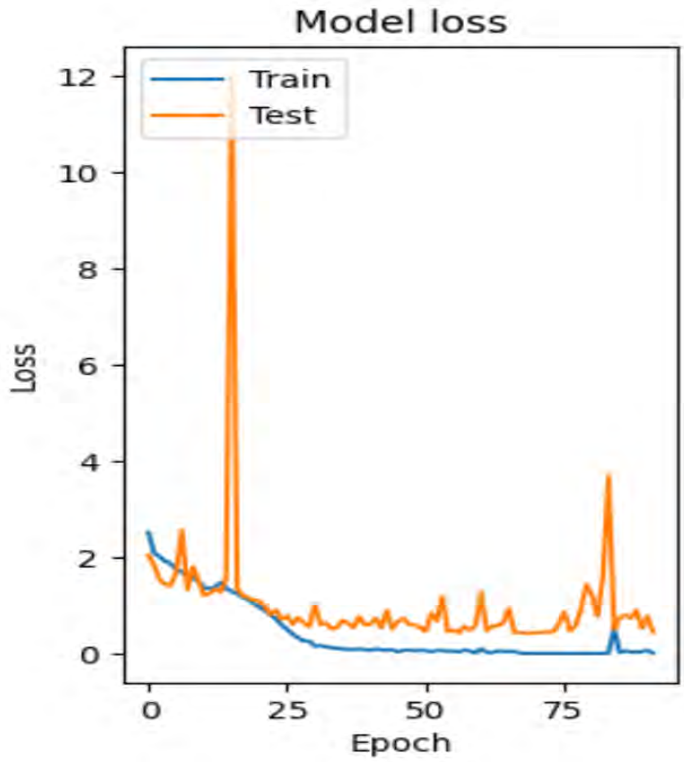


Figure 7.4: Custom Model Train-Test Loss vs Epoch Graph

7.3 Pre-Train VS Custom Model Difference

The concluding phase of our research entails the comparison of the performance between the pre-trained models, namely ResNet50, Inceptionv3, VGG16, and InceptionResNetV2, and our customized model.

ResNet50, Inceptionv3, VGG16, and InceptionResNetV2 demonstrate different levels of performance when used for the categorization of corneal ulcer grades.
Both ResNet50 and VGG16 exhibit comparable levels of accuracy, with ResNet50 having an accuracy rate of 85% and VGG16 achieving 82%.
The Inceptionv3 model exhibits a relatively low accuracy rate of 48%, indicating potential difficulties in the process of feature extraction.
The performance of InceptionResNetV2 in this particular task is found to be poor, as evidenced by a validation accuracy of merely 19.9%.
The custom-designed model exhibits a noteworthy validation accuracy of 90%, surpassing the performance of all pre-trained models.
The training accuracy of the model demonstrates a noteworthy achievement of 99%, demonstrating its ability to learn intricate patterns from the provided training dataset.
Although overfitting is observed to a limited extent, its influence on the model's performance is not considerable.

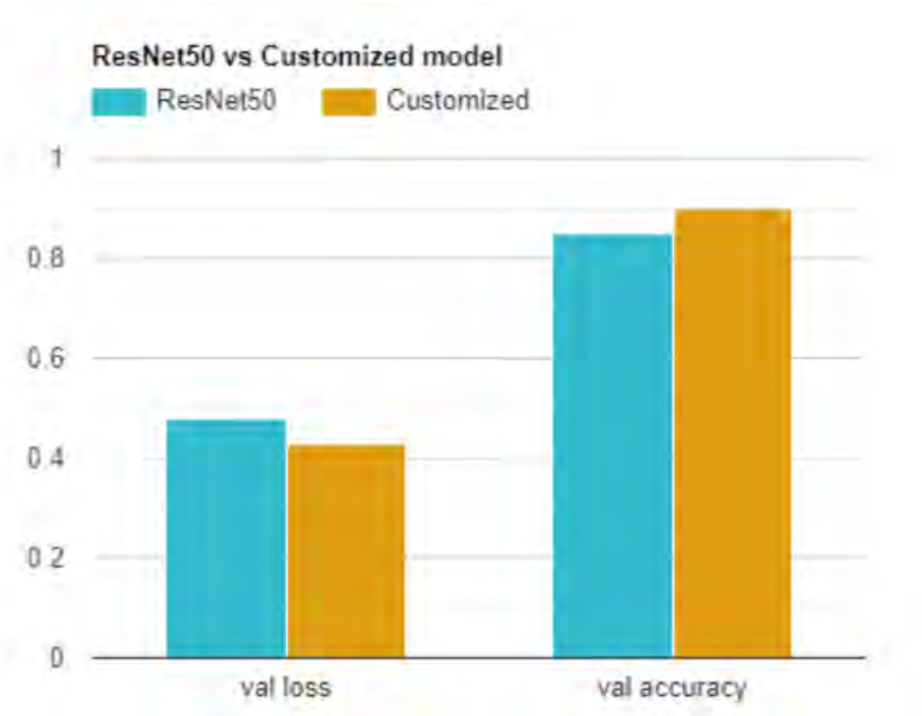


Figure 7.5: ResNet50 vs Customized Model

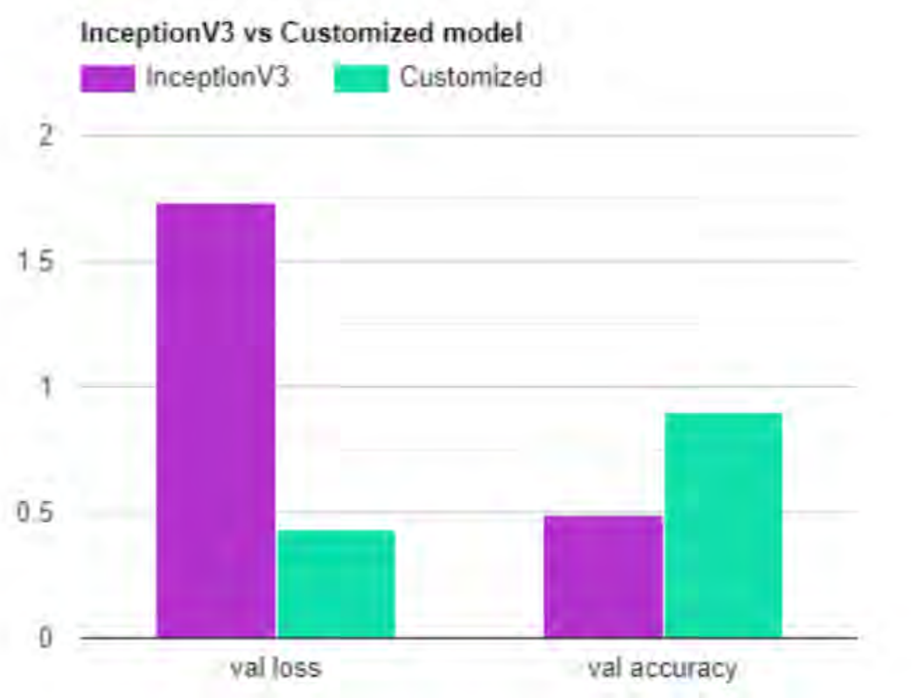


Figure 7.6: InceptionV3 vs Customized Model

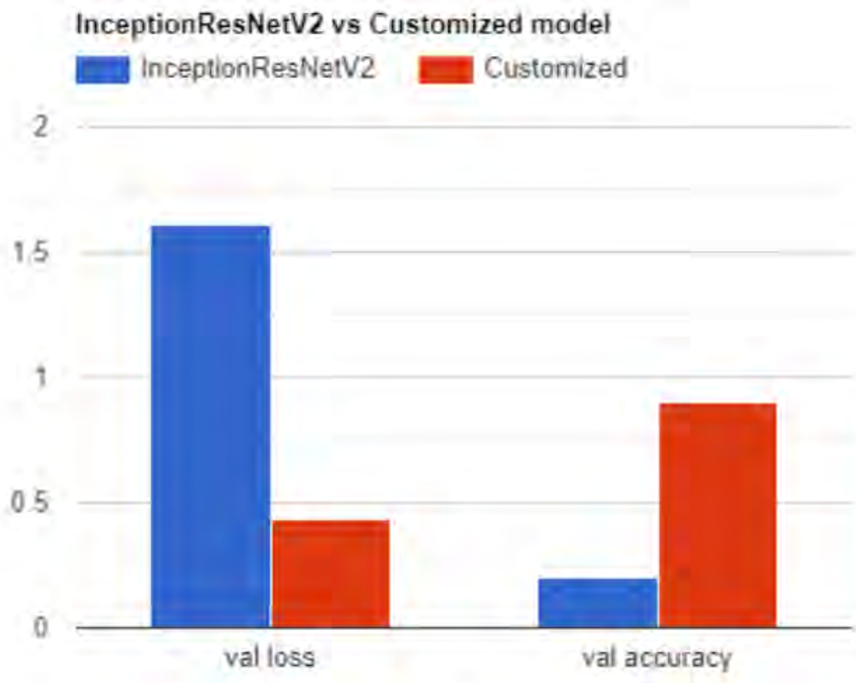


Figure 7.7: InceptionResNetV2 vs Customized Model

Figure 7.8: VGG16 vs Customized Model

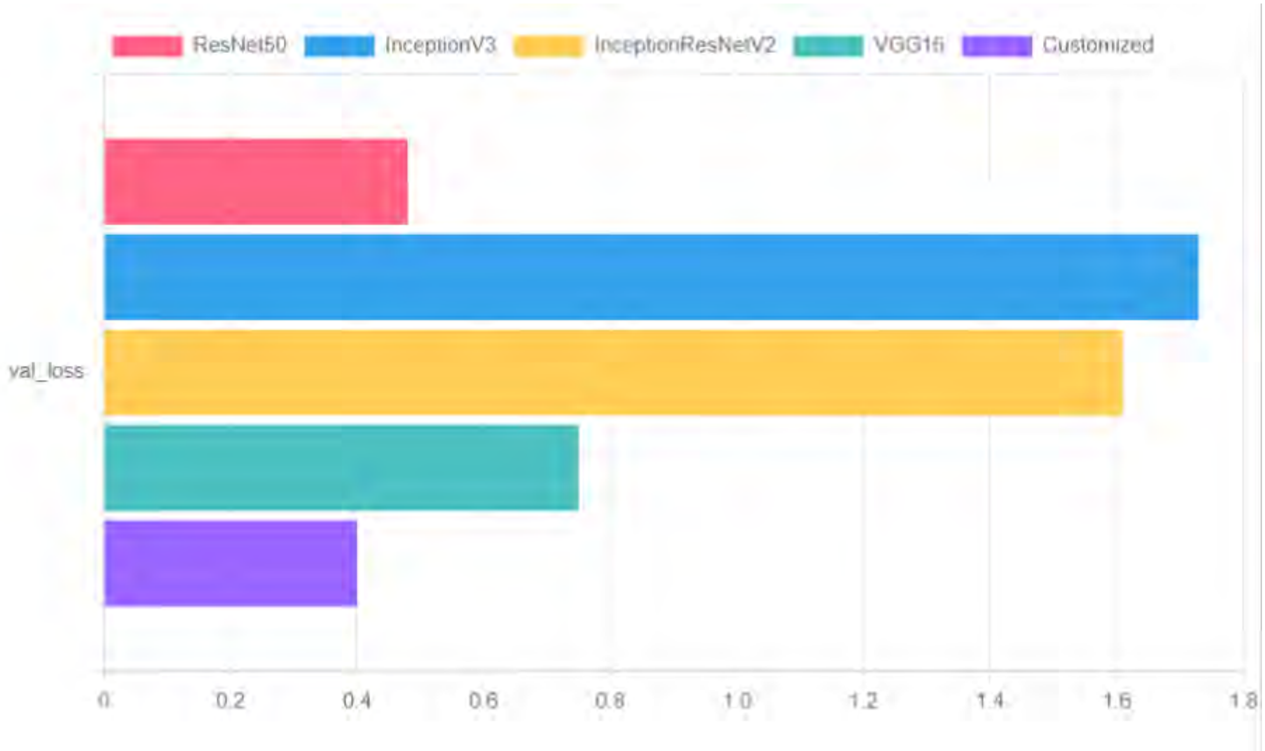


Figure 7.9: Validation Loss of Pre-trained Model vs Customized Mode

Figure 7.10: Validation Accuracy of Pre-Trained vs Customized Model

These are the graphical visualizations of validation loss and validation accuracy in between the pre-trained model and customized model. These demonstrate how our customized model outperforms pre-trained models for validation accuracy and validation loss.

Additionally, the confusion matrix is a tabular representation that provides an overview of the performance of a machine learning model when evaluated on a specific dataset. The measurement of classification model performance is frequently used for evaluating the accuracy of predicting categorical labels for input instances. There are four important variables in the confusion matrix:

- True positives (TP) refer to incidents that have been accurately predicted as positive.
- True negatives (TN) refer to occasions in which the prediction accurately identifies negative cases.
- False positives, also known as Type I errors, refer to instances where positive outcomes are incorrectly predicted.
- False negatives, also known as Type II errors, refer to instances where negative outcomes are predicted incorrectly.

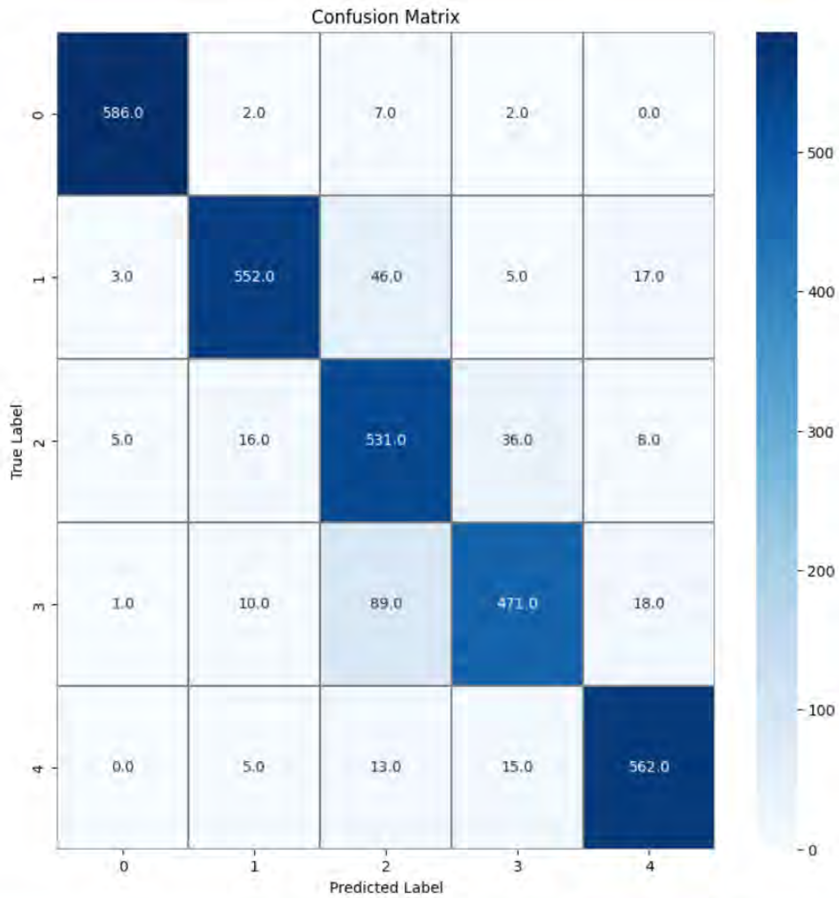


Figure 7.11: Confusion Matrix

Table 7.5: Custom Model Precision, Recall and F1-Score results for Grade classes

Grade	Precision	Recall	F1-Score	Support
Grade 0	0.98	0.94	0.96	597
Grade 1	0.98	0.89	0.93	623
Grade 2	0.77	0.89	0.83	596
Grade 3	0.89	0.80	0.84	589
Grade 4	0.93	0.94	0.94	595
Macro avg	0.90	0.90	0.90	3000
Weighted avg	0.90	0.90	0.90	3000

The Precision, Recall and F1 scores are mainly used for testing the performance matrix for evaluating the effectiveness of image classification models like pre-trained modes and custom models. Above this value are calculated for our customized model. This matrix value provides the model's ability to classify data correctly. Here, Precision is measuring the ratio of the predicted positive value to the total value predicted as positive. The precision formula is

$Precision = \frac{TruePositive}{TruePositive+FalsePositive}$

Next, Recall is also explained as sensitivity or true Positive Rate and it measures the ratio of correctly predicted positive instances to the total actual positive instances. The precision formula is

$Recall = \frac{TruePositive}{TruePositive+FalseNegative}$

Lastly, the F1 Score may be defined as the mathematical average of accuracy and recall, specifically calculated using the harmonic mean. The metric offers a trade-off between accuracy and recall, making it a valuable measure in situations where both false positives and false negatives have significant importance. The F1 Score formula is

$F1Score = \frac{2*Precision*Recall}{Precision+Recall}$

The F1 score is calculated as the harmonic mean of precision and recall, incorporating both measures in the following equation: The utilization of the harmonic mean in substitution of a simple average is justified due to its ability to penalize excessive values.

To conclude, the outcomes of our study showcase the exceptional proficiency of our model in accurately classifying corneal ulcers based on their severity. This underscores the need to develop customized model architectures that are specifically designed for accurate medical image analysis tasks.

Although pre-trained models are useful for a range of tasks, optimizing models for compatibility with specific features of the dataset can result in significant enhancements in performance and diagnostic precision.

In conclusion, the exceptional performance of our customized model highlights the promise of specialized deep-learning architectures in the field of medical image analysis, specifically in the domain of corneal ulcer grading. The results mentioned above highlight the significance of carefully choosing and optimizing models according to the particular demands and complexities of medical image datasets, hence improving the accuracy as well as reliability of healthcare diagnostics.

Chapter 8

Conclusion

8.1 Summarizing of the whole work

For the automatic classification of corneal ulcers, here a tailored deep-learning model has been proposed and created in this research paper. Pre-trained models are implemented, for instance, VGG16, ResNet50, and Inception V3, as the base architectures by using the capability of transfer learning. The attained outstanding validation accuracy rate is 90% on the dataset of 712 corneal ulcer photos after thorough testing and fine-tuning. The outcomes from the customized model demonstrate considerable promise in the area of automatically classifying medical images. Accurate detection of corneal ulcers is crucial for prompt diagnosis and treatment, which can eventually lead to better patient outcomes. The performance of this model shows how it might help doctors make well-informed judgments and improve the effectiveness of diagnosis. This research also paves the way for additional analysis and development. [9] Higher accuracy rates and more reliable predictions may result from the investigation of various pre-trained models, ensemble techniques, and hyperparameter modification. Further insights into the characteristics influencing the model's decisions may be gained by examining attention mechanisms and uncertainty estimating strategies, which will also improve the model's reliability and interpretability. Additionally, domain adaptation strategies and ongoing development and curation of the corneal ulcer dataset can aid in generalizing the model's performance to a variety of healthcare settings and patient demographics. The conducted research indicates the possibility for creative solutions in other medical areas and lays the groundwork for future breakthroughs in automated medical picture classification as the science of deep learning develops. Furthermore, this perusal has the potential to revolutionize healthcare practices all across the world with further advancements and partnerships between the medical and machine-learning communities. In the end, the study adds to the expanding corpus of research on the use of deep learning for medical picture processing. All the findings here have important ramifications for the future of diagnosing corneal ulcers and can be used as a springboard for additional investigation and improvements in automated categorization systems for a variety of medical diseases.

8.2 Future Plan

There are a number of potential directions for additional research and development based on the unique model we have created for the automatic classification of corneal ulcers. Some ideas that we can implement include learning through transfer, using different base models, self-supervised pre-training, domain adaptation, employing ensemble models, optimization of hyperparameters, strategies for data augmentation, integrating attention mechanisms, uncertainty estimation and multi-task learning, data collection and curation as well as interpretability and explainability.

Experiments can be done with using different pre-trained models as the base model, such as DenseNet, MobileNet, EfficientNet, or Xception and VGG-19. [4] Each architecture might have some advantages that could boost performance for a particular task. To pre-train the model on a large dataset of unlabeled corneal pictures before fine-tuning it on the smaller labelled sample, self-supervised learning techniques can be observed. This can aid in properly utilizing more data. Moreover, investigation of methods to modify the model so that it can function well with data from different sources or domains if it was trained on data from a specific institution or source. This will increase the model's adaptability.

This will lead to aggregate predictions from various models by using techniques like model averaging, voting, or stacking. This frequently results in stronger performance and outcomes. To determine the ideal set of hyperparameters for our model, perform a thorough hyperparameter search utilizing methods such as grid-based search, random searching, or Bayesian optimization. Trying out various data augmentation methods to observe how they affect the efficiency of models. We can look at expansions for medical photos that are domain-specific.

Incorporating self-attention or spatial attention to emphasize pertinent areas in the images will help the model better focus on key elements. Investigate uncertainty estimation approaches such as Monte Carlo Dropout, Bayesian Neural Networks, or uncertainty-aware loss functions to evaluate the model's reliability in its estimates, which is particularly relevant for medical purposes. Extend the model to carry out several related tasks concurrently, such as categorizing various corneal disorders or spotting more anomalies in corneal images. Increase the dataset's size and diversity with more representative samples over time. Better generalization and performance may result from a bigger, more complete dataset. Looking at techniques to analyze and explain the model's choices, like Grad-CAM or saliency maps, to reveal the features influencing the classification choices.

It's important to keep in mind that deep learning is a subject that is always developing new approaches and structures. In order to use cutting-edge techniques for the automated classification of corneal ulcers, we need to stay current on the most recent research and technological developments. The mix of creativity, subject-matter expertise, and a readiness to consider novel concepts will open the door for fascinating new research in this field.

Bibliography

[1] R. A. Hyndiuk, E. L. Kazarian, R. Schultz, and S. Seideman, "Neurotrophic corneal ulcers in diabetes mellitus," *Archives of Ophthalmology*, vol. 95, no. 12, pp. 2193–2196, 1977.

[2] L. O. Chua and T. Roska, "The cnn paradigm," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 40, no. 3, pp. 147–156, 1993.

[3] M. Sonka, V. Hlavac, R. Boyle, M. Sonka, V. Hlavac, and R. Boyle, "Image pre-processing," *Image processing, analysis and machine vision*, pp. 56–111, 1993.

[4] M. Boivin and S. Hymel, "Peer experiences and social self-perceptions: A sequential model.," *Developmental psychology*, vol. 33, no. 1, p. 135, 1997.

[5] M. Srinivasan, C. A. Gonzales, C. George, *et al.*, "Epidemiology and aetiological diagnosis of corneal ulceration in madurai, south india," *British Journal of Ophthalmology*, vol. 81, no. 11, pp. 965–971, 1997.

[6] H.-J. Chen, R. T. Pires, and S. C. Tseng, "Amniotic membrane transplantation for severe neurotrophic corneal ulcers," *British Journal of Ophthalmology*, vol. 84, no. 8, pp. 826–833, 2000.

[7] C. R. Sanders and K. Oxenoid, "Customizing model membranes and samples for nmr spectroscopic studies of complex membrane proteins," *Biochimica et Biophysica Acta (BBA)-Biomembranes*, vol. 1508, no. 1-2, pp. 129–145, 2000.

[8] M. Upadhyay, P. Karmacharya, S. Koirala, *et al.*, "The bhaktapur eye study: Ocular trauma and antibiotic prophylaxis for the prevention of corneal ulceration in nepal," *British Journal of Ophthalmology*, vol. 85, no. 4, pp. 388–392, 2001.

[9] B. Behkamal, M. Kahani, and M. K. Akbari, "Customizing iso 9126 quality model for evaluation of b2b applications," *Information and software technology*, vol. 51, no. 3, pp. 599–609, 2009.

[10] F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Sequential model-based optimization for general algorithm configuration," in *Learning and Intelligent Optimization: 5th International Conference, LION 5, Rome, Italy, January 17-21, 2011. Selected Papers 5*, Springer, 2011, pp. 507–523.

[11] F. M.-T. et al. (2012), [Online]. Available: https://www.hindawi.com/journals/ipid/2012/851563/?fbclid=IwAR1TfM1bMjMjS54m2KFkRF4-gJFgjQNKsJN_PWFV7rrIp1zIdHeMzwuqEEI (visited on 09/30/2010).

[12] G. Amescua, D. Miller, and E. Alfonso, "What is causing the corneal ulcer? management strategies for unresponsive corneal ulceration," *Eye*, vol. 26, no. 2, pp. 228–236, 2012.

[13] C. Lam, D. Yi, M. Guo, and T. Lindsey, "Automated detection of diabetic retinopathy using deep learning," *AMIA summits on translational science proceedings*, vol. 2018, p. 147, 2018.

[14] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Sep. 2018.

[15] H. Qassim, A. Verma, and D. Feinzimer, "Compressed residual-vgg16 cnn model for big data places image recognition," in *2018 IEEE 8th annual computing and communication workshop and conference (CCWC)*, IEEE, 2018, pp. 169–175.

[16] M. Shaha and M. Pawar, "Transfer learning for image classification," in *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 2018, pp. 656–660. DOI: 10.1109/ICECA.2018.8474802.

[17] A. Akram and R. Debnath, "An efficient automated corneal ulcer detection method using convolutional neural network," in *2019 22nd International Conference on Computer and Information Technology (ICCIT)*, IEEE, 2019, pp. 1–6.

[18] T. Inc. "Title of webpage." (2019), [Online]. Available: %5Curl%7Bhttps://docs.ecognition.com/v9.5.0/eCognition_documentation/Reference+Book/23+Convolutional+Neural+Network+Algorithms/Convolutional+Neural+Network+Algorithms.html%7D.

[19] J. Y. Kim, H. E. Lee, Y. H. Choi, S. J. Lee, and J. S. Jeon, "Cnn-based diagnosis models for canine ulcerative keratitis," *Scientific reports*, vol. 9, no. 1, p. 14 209, 2019.

[20] I. Z. Mukti and D. Biswas, "Transfer learning based plant diseases detection using resnet50," in *2019 4th International conference on electrical information and communication technology (EICT)*, IEEE, 2019, pp. 1–6.

[21] R. Sayres, A. Taly, E. Rahimy, *et al.*, "Using a deep learning algorithm and integrated gradients explanation to assist grading for diabetic retinopathy," *Ophthalmology*, vol. 126, no. 4, pp. 552–564, 2019.

[22] L. Deng, J. Lyu, H. Huang, Y. Deng, J. Yuan, and X. Tang, "The sustech-sysu dataset for automatically segmenting and classifying corneal ulcers," *Scientific data*, vol. 7, no. 1, p. 23, 2020.

[23] T. R. Gadekallu, N. Khare, S. Bhattacharya, S. Singh, P. K. R. Maddikunta, and G. Srivastava, "Deep neural networks to predict diabetic retinopathy," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–14, 2020.

[24] H. Gu, Y. Guo, L. Gu, *et al.*, "Deep learning for identifying corneal diseases from ocular surface slit-lamp photographs," *Scientific reports*, vol. 10, no. 1, p. 17 851, 2020.

[25] D. Theckedath and R. Sedamkar, "Detecting affect states using vgg16, resnet50 and se-resnet50 networks," *SN Computer Science*, vol. 1, pp. 1–7, 2020.

[26] F. M.-T. et al. (2021), [Online]. Available: https://www.hindawi.com/journals/joph/2021/3940151/?fbclid=IwAR3GGEWh1KQcptZdppRbyQgelZocqSVUTfyHhgq EjZX7MEqZo98 (visited on 09/30/2010).

[27] A. Ayala, T. Ortiz Figueroa, B. Fernandes, and F. Cruz, "Diabetic retinopathy improved detection using deep learning," *Applied Sciences*, vol. 11, no. 24, p. 11 970, 2021.

[28] H. NLM NIH. "National library of medicine." (2021), [Online]. Available: https://www.ncbi.nlm.nih.gov/?fbclid=IwAR0rHqhEUdWKPdQ3fO2yQowgoMvxDXW20UP (visited on 09/30/2010).

[29] A. Vidhya. "Building a custom cnn model: Identification of covid-19." (2021), [Online]. Available: https://www.analyticsvidhya.com/blog/2021/12/cnn-model-covid-19/ (visited on 09/30/2010).

[30] L. Lv, M. Peng, X. Wang, and Y. Wu, "Multi-scale information fusion network with label smoothing strategy for corneal ulcer classification in slit lamp images," *Frontiers in Neuroscience*, vol. 16, p. 993 234, 2022.

[31] I. A. Qasmieh, H. Alquran, A. Zyout, Y. Al-Issa, W. A. Mustafa, and M. Alsalatie, "Automated detection of corneal ulcer using combination image processing and deep learning," *Diagnostics*, vol. 12, no. 12, p. 3204, 2022.

[32] C. .-. G. 2023. "Cnn image classification in tensorflow with steps examples." (2023), [Online]. Available: https://www.guru99.com/convnet-tensorflow-image-classification.html (visited on 09/30/2010).

[33] 2. PyImageSearch. "Keras imagedatagenerator and data augmentation." (2023), [Online]. Available: https://pyimagesearch.com/2019/07/08/keras-imagedatagenerator-and-data-augmentation/ (visited on 09/30/2010).

[34] satori. "Data generalization: The specifics of generalizing data." (2023), [Online]. Available: https://satoricyber.com/data-masking/data-generalization/#:~:text=Data%20generalization%20is%20the%20process%20of%20summarizing%20data%20by%20replacing,numbers%20with%20higher%2Dlevel%20concepts. (visited on 09/30/2010).

[35] R. GmbH. "Researchgate." (2008-2023), [Online]. Available: %5Curl%7Bhttps://www.researchgate.net/figure/List-of-formulas-for-the-operations-in-the-CNN-Operation-Formula_tbl1_338431622%7D.