Solutions to the unseen domain problem for machine learning-based glaucoma detection from retinal fundus images

A thesis

submitted in partial fulfilment

of the requirements for the Degree

of

Doctor of Philosophy in Computer Science

at

The University of Waikato

by

H. N. Gunasinghe

THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

2024

Abstract

Ocular diseases are a significant problem faced by 70 million people worldwide yearly. Identifying an eye disease requires a medical professional with years of specialist training. Glaucoma screening is often done manually, which is time-consuming. Automatic eye disease identification could be introduced to clinics to help speed up the detection and treatment of eye diseases. Retinal fundus images are helpful for glaucoma screening. However, the existing glaucoma identification systems trained using images from a specific camera fail to perform well with those captured from a new camera. This research aims to find solutions to address the above-unseen domain problem in machine learning-based glaucoma detection systems from retinal fundus images. The study was conducted in three phases, including an initial study that identifies the problem domain, followed by the manipulation of image preprocessing and image augmentation techniques to produce a more generalised glaucoma detection system.

In the first stage, twenty-eight pre-trained deep learning models for object recognition tasks were compared as potential feature extractors for glaucoma classification from retinal fundus images of the REFUGE dataset. First, the images were automatically cropped around the optic nerve head using a template-matching algorithm. Features were extracted using the pre-trained networks from both whole and cropped images. An extended feature set was created by concatenating those two feature sets. Finally, a ten-fold cross-validation experiment was conducted to compare the performance of random forest and logistic regression classifiers against each feature set. The best setup was when features were extracted from images using the ResNet101V2 ImageNet-pre-trained neural network and classified using a random forest classifier. However, the accuracy dropped when testing it with images from a new camera. Hence, the study was directed towards the unseen domain problem in the second stage.

In the study's second phase, transfer learning-based domain generalisation was applied together with multiple image preprocessing methods: input standardisation, median filtering and multi-image histogram matching for glaucoma detection using retinal fundus images from multiple cameras. The anal-

ysis included images from the RIMONEr2 and REFUGE datasets, which were captured using three camera models. A set of experiments were conducted using all possible combinations of training and testing camera devices, using the best system found in the first stage. Images were preprocessed in six different ways using either a single or a combination of three different preprocessing methods to see their effect on generalisation. The results indicated that the stylisation of test data might lead to better generalisation while reducing the retraining of an existing system.

As a result, we compared multi-image histogram matching with neural style transferring to identify the classification accuracy during the testing phase of a model. We trained a random forest classifier and an XGBoost classifier with AlexNet and ResNet101V2 as feature extractors and tested the system following the same strategy as in phase two. Comparative results indicated that the neural style transferring better predicts the labels for unseen images. We continued experiments with neural style transferring to test publicly available models trained on the ACRIMA dataset. The method results better when reference images are selected from the same class. Given that the class information of real clinical data is unavailable, we suggest possible strategies for choosing better reference images.

Overall, this study provides solutions to develop robust machine learning systems that require no retraining with new fundus cameras. The experimental results indicate that the proposed combination of preprocessing methods can be successfully utilised for better domain generalisation in the context of different retinal fundus camera devices. Furthermore, test-time data augmentation with neural style transferring leads to better predictions for images taken from unseen retinal fundus cameras. This reduces model retraining and increases the reusability of a pre-trained machine learning-based glaucoma detection system.

Acknowledgements

I thank my supervisors, Dr Anthony Smith, Dr Micheal Mayo, Dr Abigail Koay and Dr James McKelvie, for their precious teachings. They guided and encouraged me to tackle scientific problems with great enthusiasm. Their contribution to my personal growth in scientific research is invaluable. I hope to repay their help with my contributions in future work on challenging problems. I thank my colleagues for providing reviews with relevant comments and observations, which helped me improve the thesis's quality. I thank the coordinator of the graduate study program and the members of my thesis committee.

I am thankful for the PhD scholarship from the AHEAD project in Sri Lanka, funded by World Bank. It covered all expenses for three years, including tuition and living costs. I needed it to be able to complete the degree on time. Also, my head of the department, Department of Computing, Sabaragamuwa University of Sri Lanka, for granting me study leave, processing progress reports on time and encouraging words during the study period.

Also, I want to convey my utmost gratitude to my parents, family and friends for encouraging me. It would not have been possible to complete this project without the help and support of my dearest ones.

Bibliographic Notes

The main content in this thesis has been published in peer-reviewed conferences and journals. The list of papers is as follows:

• Comparison of Pretrained Feature Extractors For Glaucoma Detection, International Symposium on Biomedical Imaging (ISBI), 2021

• Automated detection of glaucoma from retinal fundus images using a variety of fundus cameras, Journal of Clinical and Experimental Ophthalmology, 2022 [Abstract]

• Domain Generalisation for Glaucoma Detection in Retinal Images from Unseen Fundus Cameras, Asian Conference on Intelligent Information and Database Systems (ACIIDS), 2022

Additionally, some of the findings were presented at the following symposium:

• Machine learning-based glaucoma detection using fundus images from multiple cameras, New Zealand Save Sight Society Symposium, 2022

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

Ophthalmology is an area of medicine concerning the diagnosis and treatment of ocular diseases and disorders[1]. The most common ocular diseases are macular degeneration, cataracts, diabetic retinopathy and glaucoma [2]. Diabetic patients are at high risk of having vision problems, which are sometimes irreversible [3].

1.1.1 Glaucoma

Despite glaucoma being identified as a priority eye disease by the World Health Organisation [4], it is the most common reason for irreversible blindness[5] which implies that the glaucoma is not curable unless detected early. According to the statistics, an estimation of 80 million persons had glaucoma, with a predicted increase to 111 million people in 2040 [6]. Unfortunately, half of the glaucoma patients are left undetected because glaucoma is mostly asymptomatic[5]. Glaucoma is associated with irreversible, progressive vision loss and typically remains asymptomatic until late in the disease process.

There are several different sub-types of glaucoma that all result in loss of optic nerve fibres, changes in the appearance of the optic nerve head (ONH),

[1] `http://www.mrcophth.com/Historyofophthalmology/Introductory.htm`

[2] `https://www.nei.nih.gov/learn-about-eye-health/eye-conditions-and-diseases`

also known as the optic disc (OD), and corresponding visual field defects. A child from birth to 10 years may be detected with congenital glaucoma, while ten to 35 years old may have juvenile glaucoma. According to a person's age the third type of glaucoma is the adult type, which occurs in adults over 35. Various causes may define primary or secondary glaucoma. Primary causes are non-identifiable that occur in susceptible individuals. Other causes, such as trauma, drugs, other ocular diseases, intraocular surgeries, etc., cause secondary glaucoma. There are open-angle glaucoma and angle closure glaucoma according to the site of obstruction of the drainage system of the eye[3].

High intraocular pressure is a significant risk factor associated with glaucoma progression. Lowering intraocular pressure using medicines, surgery, or laser treatment can contain further visual loss in most cases [7]. Careful lifelong monitoring of adherence to treatment, intraocular pressure, optic nerve appearance and nerve fibre loss is essential to prevent loss of vision [4].

The irreversible and progressive nature of vision loss, as seen in patients with glaucoma, makes early diagnosis, close monitoring, and effective treatment critical. At present, diagnosis and monitoring of glaucoma progression rely on a highly-trained ophthalmologist completing detailed regular clinical examinations, visual field tests and imaging of the ONH. As glaucoma prevalence rises, it is increasingly challenging for many resource-constrained ophthalmic clinics to keep pace with growing demand, and many patients are at risk of vision loss [7, 8, 9]. With the increasing cohort of patients with glaucoma, in many cases, the rate-limiting factor for effective glaucoma monitoring is the availability of clinicians with the appropriate level of expertise to detect progression [10].

1.1.2 Glaucoma detection

Clinicians perform four basic tests to diagnose glaucoma. The first one is measuring intraocular pressure using a tonometer, also called tonometry. The

[3] `https://www.glaucomapatients.org/basic/statistics/`

second is the gonioscopy, which will watch the drainage angle or trabecular meshwork. It helps detect open or closed-angle glaucoma. The third test examines the optic nerve's structure, while the fourth evaluates the optic nerve's functionality with respect to the visual field or perimetry.

There are additional tests that they can perform. Pachymetry, also known as corneal thickness, helps interpret eye pressure measurements. Also, different aspects of visual function can be assessed using special visual field tests such as frequency doubling and short wave perimetry. Furthermore, imaging the optic nerve and retinal nerve fibre layer with machines helps assess and quantify the presence of glaucomatous structural damage. Additional evaluation of the angle can be done with Ultrasound biomicroscopy (UBM) / anterior segment optical coherence tomography (AS-OCT). However, these exams are required in just a few instances, as a gonioscopy exam is usually enough to evaluate the angle.

Glaucoma outset and progression can be detected by assessing changes in the appearance of the ONH in a sequence of images. The optic nerve comprises around 1.2m nerve fibres that transfer visual information to the brain. The nerve fibres are observable at the ONH as a periphery of pale tissue surrounding a central depression, the cup. The cup-to-disc ratio (CDR) indicates the severity of glaucoma [11]. One can calculate CDR by dividing the diameter of the cup by the diameter of the disk. If the CDR is below 0.3, the eye is normal. Mild glaucoma can be seen if CDR ranges from 0.4 to 0.7. If CDR is greater than 0.7, it is considered to have moderate/severe glaucoma [12].

Manual observation of the ONH and estimation of the CDR is time-consuming, inaccurate and biased [11]. Furthermore, it requires years of experience to be an expert in detecting glaucoma manually. Moreover, CDR should be interpreted in conjunction with other clinical findings and tests to accurately determine the severity of the condition, as there are a few sub-types of glaucoma where the CDR is misleading regarding severity [13]. These challenges

limit the number of patients to be monitored in a day. Hence, doctors seek the assistance of automatic tools to help with their decisions.

1.1.3 Computer-aided glaucoma detection

Automated glaucoma screening and monitoring would save time and money, enabling more efficient clinical workflow and better using limited resources. Early detection of glaucoma with the aid of an automated system would enable early treatment and prevention of irreversible visual loss for patients with glaucoma. An automated system can utilise the expertise of multiple ophthalmologists to produce accurate and repeatable results [11]. With the extra help from an automated system, the practitioner will be able to see more patients in a given time. This may even reduce the number of experts needed in a clinic.

However, the deployment of such technologies is not without significant challenges. Questions regarding the accuracy and reliability of automated systems persist, primarily due to the necessity for extensive and diverse datasets for training. Models poorly trained or trained on non-representative data can produce biased or inaccurate outputs, potentially leading to misdiagnoses [14]. Additionally, relying too much on automated systems can reduce the importance of human supervision in diagnosing diseases. This might limit the comprehensive clinical judgment that seasoned medical professionals offer.

Regulatory and ethical considerations are also essential. Automated systems in healthcare must navigate rigorous validation processes to satisfy safety and efficacy standards imposed by healthcare authorities [15]. Ethical issues, such as ensuring patient data privacy and addressing the implications of diagnostic errors, must also be addressed carefully.

All in all, while computer-aided glaucoma detection systems offer promising advancements in eye care, their implementation should be approached with a balanced perspective that emphasises support rather than replacement of human expertise. Such technologies need to be developed and utilised within

a framework that addresses technical, ethical, and practical challenges to fully capitalise on their potential while safeguarding against risks. The goal should be to enhance, rather than replace, the capabilities of healthcare professionals in delivering patient-centered care [16].

1.2 Research problem statement

There are potential issues when deploying automatic detectors in practical clinical settings due to various problems, such as device dependency of data. In this research, we are keen to explore the current systems and their issues, the impact of device variability in automated glaucoma detection and probable approaches to mitigate the issues. Working towards this, we have formulated the research questions as follows.

1. What are the best machine learning-based feature extractors and useful features in glaucoma detection from retinal fundus images?

2. How to improve the generalisation of machine learning models for new fundus cameras in glaucoma classification through image preprocessing?

3. Will applying neural style transferring during the test time on pre-trained machine learning systems improve the generalisation for various fundus cameras?

1.3 Research goal

Camera bias plays a significant role inadvertently in machine learning research involving images. It is dangerous in medical domains because of the implicit assumption that the test images come from the same distribution (i.e. same camera) as the training images. This research aims to reduce the unseen domain problem of deep learning-based glaucoma classification using retinal images from various fundus cameras and improve accuracy while saving processing time and reducing the computational effort.

1.4 Research objectives

The following objectives are formulated in order to achieve the above goal.

1. Compare available deep learning-based feature extractors and identify useful features for glaucoma classification

2. Explore the problem of camera dependency of machine learning-based glaucoma detection using retinal fundus images with respect to colour and spatial features

3. Experiment on transfer learning-based domain generalisation along with a combination of image preprocessing techniques and conventional data augmentation for glaucoma detection

4. Experiment on applying neural style transferring to improve the reduced accuracy in glaucoma detection caused by a change in the device after model training

1.5 Contributions of the Thesis

The findings of this study are effective for the benefit of the patient, considering that glaucoma is a severe eye condition that may lead to irreversible blindness. The greater demand for automated tools by eye clinics justifies the need for more effective and efficient systems using state-of-the-art (SOTA) technologies such as artificial intelligence. Numerous manufacturers produce a diverse range of fundus camera models that incorporate different image-capturing technologies, including digital and laser-based methods. Additionally, other imaging technologies, such as Optical Coherence Tomography (OCT) photographs, are employed in the diagnosis of glaucoma. Due to ethical concerns surrounding the distribution of personal data, these systems are often restricted to use within a single clinic. Consequently, there is a significant need to expand the availability of these diagnostic systems.

Thus, clinics that apply the recommended approach derived from the results of this study will be able to effectively adapt and reuse a system that is already available. Otherwise, they can develop a machine learning system that generalises better on newly introduced fundus cameras such as laser-based ones. Overall, the investigation will uncover plausible solutions to the unseen domain problem in glaucoma detection from retinal fundus images by machine learning-based computer systems that many researchers could not explore. Thus, a new, more generalised glaucoma detection system may be developed.

Finally, the solutions will help early detection of glaucoma and reduce the time-consuming diagnosis process. With the help of automated tools, clinics can reduce the waiting time for patients to be diagnosed, thereby increasing the chances of early detection. For example, in an underdeveloped country with limited resources, such as expert knowledge and the latest technologies, clinics can use a handheld fundus camera to capture an image and send it to a system for initial diagnosis. In many cases, the cost of treatment is much higher than the cost of diagnosis.

1.6 Plan of the Thesis

The research undertaken results in this thesis with the following chapters:

Introduction

The first chapter gives the background of the research. Starting with conventional and computer-aided glaucoma identification, it describes the unseen domain problem caused by device variability. In particular, a list of currently used devices is listed along with publicly available datasets from several devices and sample images are included to show the visible differences between them. Following this, a research problem statement is formulated. Finally, the research aims and objectives are given.

Literature Review

The literature review chapter studies recent literature based on the unseen domain problem and glaucoma classification to identify the research gaps. It describes the concepts of machine learning/ transfer learning, unseen domain vs. domain shift vs. out-of-distribution, and domain adaptation to differentiate between terminology. Furthermore, a description of the datasets used in this research is included to show their diversity. Finally, we compare and contrast the related work, their performance and limitations to summarise the proposed approach.

Problem identification

This chapter explains the initial study undertaken to identify the usability of transfer learning in glaucoma classification. The study could find the best feature extractor among multiple pre-trained models using cross-validation. Furthermore, it illustrates the problem of the unseen domain when a model is trained on the devices from a particular device and tested on another device. The initial work was extended to compare with the latest models, such as InceptionV3.

Comparison of preprocessing methods against unseen domain problem

This chapter covers one of this study's primary objectives: identifying suitable preprocessing methods to reduce domain variability. Here, we compare three methods: median filtering, input standardisation and histogram matching. Importantly, we introduce randomised multi-image histogram matching that changes an image to appear differently, creating an output similar to neural style transferring found in domain adaptation. Furthermore, we demonstrate that a mixture of preprocessing methods improves the generalisability of a model to camera variability. The algorithms were initially tested using the

networks identified by the preliminary study and extended later to be tested using XGBoost, one of the latest methods found in recent literature.

Surrogate optimisation of deep stack transformations

Augmentation is another generalisation method in contrast to image preprocessing. This chapter investigates the augmentation method called deep stack transformations (DST) initially tested with MRI segmentation. The method applies N image transformations sequentially per given input image; each transformation has a magnitude and a probability of application. We use DST for glaucoma image classification. Furthermore, we optimise parameters using the surrogate optimisation method (SurrogateRBF-DST) and compare it with local search (LS-DST) optimisation. Additionally, we design the network training and testing process in a novel way where we choose train, validation and test images from three different devices. Finally, we achieved improved generalisation across devices while determining optimal parameters.

Neural style transfer for domain generalisation

This chapter works with another primary goal of the research: the application of data augmentation during the testing time of a pre-trained machine learning model. We used models trained on ACRIMA images labelled according to the glaucoma class, and both are publicly available. Here, we styled the images of test/target data from a separate camera using the reference images from the ACRIMA dataset. Tested on public datasets, selecting the reference images from the same class as the target images gives the best classification accuracy than classifying over the original images of the target dataset. Also, the worst accuracy was given when the reference and target images were from different classes. However, label information is unavailable in practical situations, which is challenging unless an ophthalmologist assigns an initial label. Hence, we recommend strategic algorithm development for choosing better reference images for styling.

Conclusions, contributions and future work

The last chapter includes the conclusions, contributions and future research directions raised from the study. We highlight the challenges and limitations of the research as well.

Chapter 2

Background

2.1 Machine learning and transfer learning

Transfer learning (TL) is a research problem in machine learning (ML) that focuses on reserving knowledge gained while solving one problem and applying it to another different but related problem [17]. From the practical standpoint, reusing or transferring information from previously learned tasks to new ones can significantly improve the sample efficiency of a learning model [18].

The definition of transfer learning can be given using the terms: domains and tasks. A domain $\mathcal{D}$ consists of a feature space $\mathcal{X}$ and a marginal probability distribution $P(X)$, where $X = x_1, ..., x_n \in \mathcal{X}$. Given a specific domain, $\mathcal{D} = \mathcal{X}, P(X)$, a task consists of two components: a label space $\mathcal{Y}$ and an objective predictive function $f : \mathcal{X} \rightarrow \mathcal{Y}$. The function f is used to predict the corresponding label $f(x)$ of a new instance x. This task, denoted by $\mathcal{T} = \mathcal{Y}, f(x)$, is learned from the training data consisting of pairs x_i, y_i, where $x_i \in X$ and $y_i \in \mathcal{Y}$ [19].

Given a source domain $\mathcal{D}_S$ and learning task $\mathcal{T}_S$, a target domain $\mathcal{D}_T$ and learning task $\mathcal{T}_T$, where $\mathcal{D}_S \neq \mathcal{D}_T$, or $\mathcal{T}_S \neq \mathcal{T}_T$, transfer learning aims to help improve the learning of the target predictive function $f_T(\cdot)$ in $\mathcal{D}_T$ using the knowledge in $\mathcal{D}_S$ and $\mathcal{T}_S$ [19].

2.2 Unseen domain problem

A domain shift [20], also known as distribution shift [21], is a difference between the data distributions of an algorithm's training dataset and the dataset it finds when deployed. Traditional machine-learning algorithms usually adjust poorly to domain shifts. Hence, domain adaptation techniques have become popular among the modern machine-learning community [20].

Domain adaptation [22] is an area associated with machine learning and is a subcategory of transfer learning. Domain adaptation can be defined as the ability to apply an algorithm trained in one (or more) "source domain(s)" to a different (but related) "target domain". In domain adaptation, the source and target domains all have the same feature space (but different distributions); however, there may be examples where the target domain's feature space differs from source feature space(s) [23]. This technique is commonly used in the medical domain because labelled images are unavailable in clinical scenarios. Machine learning techniques such as generative adversarial networks are widely used in such situations.

In general, the out-of-distribution (OOD) generalisation problem can be defined as a representation by an instance of a supervised learning problem where the test distribution $\mathcal{P}_{te}(X, Y)$ shifts from the training distribution $\mathcal{P}_{tr}(X, Y)$ and stays unexplored during the training stage [24]. General machine learning systems assume that the training and testing data come from the same distribution or that the data are independently and identically distributed [25]. However, in practical situations, the distribution of test data may differ from the training data distribution $\mathcal{P}_{tr}(X, Y) \neq P_{te}(X, Y)$. Furthermore, OOD can be considered a special case of domain adaptation because OOD is considered when the labelling of both training and test data is known. In contrast, domain adaptation can be known or unknown in test data labels.

OOD scenarios can be encountered in routine clinical practice where new imaging devices are introduced in the clinical workflow [26]. This is caused by the spatial evolution of data making machine learning lacking in practice [24].

It implies that model retraining would be required to maintain the accuracy of OOD scenarios. Generalisation between imaging devices is a specific instance of domain generalisation dedicated to addressing the problem of domain shift caused by device-dependent data properties into machine learning models.

2.3 Retinal fundus imaging

Retinal fundus imaging is one of the least invasive methods in diagnosing glaucoma. The differences caused by the fundus cameras have important implications for screening eye diseases when used with deep learning [27].

A fundus camera is a specialised low-power microscope attached to a camera. It is optically designed based on the indirect ophthalmoscope. The optical angle of acceptance of the lens, also known as the angle of view, is used to describe a fundus camera. A normal angle view camera with an angle of 30° creates an image 2.5 times larger than normal. Wide-angle cameras can capture images between 45° and 140° to provide proportionately less retinal magnification. The angle of view is 20° or less in a narrow-angle fundus camera. Simultaneous stereo fundus cameras place two images side by side on a single 35mm frame using a single exposure [28].

Fundus photographs visually record the current ophthalmoscopic appearance of a patient's retina. They are routinely used in a wide variety of ophthalmic conditions, such as glaucoma (increased pressure in the eye), which can damage the optic nerve over time. The physician studies subtle changes in the optic nerve and then recommends the appropriate therapy, using serial fundus photographs [29].

2.3.1 Commercially available fundus cameras

We conducted an extensive background study on available table-top retinal fundus cameras worldwide. The aim was to identify the manufacturers, device models, their technology and publicly available datasets that consist of

images from each device. There are around 30 different retinal fundus cameras, mainly from nine manufacturers. However, only seven public glaucoma-labelled datasets contain their camera information. Table 2.1 includes the list of manufacturers and publicly available glaucoma datasets.

Table 2.1: List of fundus camera manufacturers and publicly available glaucoma labelled datasets with device information (Key: 1 = `https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/1YRRAC`)

Manufacturer	Datasets	Dataset URL	Image count	Glaucoma	Non Glaucoma
Canon	REFUGE01 (Test/Val)	https://refuge.grand-challenge.org/	800	80	720
	High-Resolution Fundus Quality Assessment	https://www5.cs.fau.de/research/data/fundus-images/	30	15	15
NIDEK	RIMONEr2	http://medimrg.webs.ull.es//	455	200	255
	Glaucoma Fundus	https://bit.ly/3I36fz5[1]	1542	756	786
Topcon	ACRIMA	https://figshare.com/s/c2d31f850af14c5b5232/	705	396	309
ZEISS	REFUGE01 (Train)	https://refuge.grand-challenge.org/ /	400	40	360
	INSPIRE	https://medicine.uiowa.edu/eye/inspire-datasets /	40	40	-
EasyScan	No publicly available glaucoma datasets				
Forus Health					
Kowa Optimed					
S4Optik					
Volk					

2.4 Publicly available retinal fundus image datasets

In this study, we evaluate the accuracy of glaucoma detection algorithms with respect to different fundus camera devices. Images from three distinct fundus cameras from different manufacturers, namely Canon CR-2, NIDEK AFC-210 and ZEISS VISUCAM-500, are used in our work.

2.4.1 REFUGE dataset

REFUGE dataset is one of the largest glaucoma datasets in public [30]. It contains retinal fundus images labelled as glaucoma and non-glaucoma. The dataset was created in 2018 and has an updated version in 2020. Two latest fundus camera models, namely, Zeiss Visucam 500 retinal fundus camera and Canon CR-2 camera, were used to capture the images having the resolution of 2124×2056px and 1634×1634px respectively. The images focus on the posterior pole, ensuring visibility of both the macula and the optic disc. In each image, the ONH is aligned to the left. The whole dataset consists of 1200 with 120 glaucoma images and 1080 non-glaucoma images making it a hugely unbalanced dataset. Moreover, 400 training images were captured using a Zeiss camera, and the remaining 800 images were captured using a Canon camera. The new version has additional 400 images captured using a third camera, but the camera information and the labels are hidden from the public.

2.4.2 RIMONEr2 Dataset

The RIMONE dataset was created in 2011 in three versions. The second version (RIMONEr2), introduced in 2014, has the largest number of images totalling 455: 200 images from glaucoma patients and 255 images from subjects with no glaucoma. A Nidek AFC-210 fundus camera captured these fundus photographs with a body of Canon EOS 5D Mark II. The settings include a vertical and horizontal field of view of 45°. All images were manually segmented by a glaucoma specialist [31]. The dataset has images cropped around

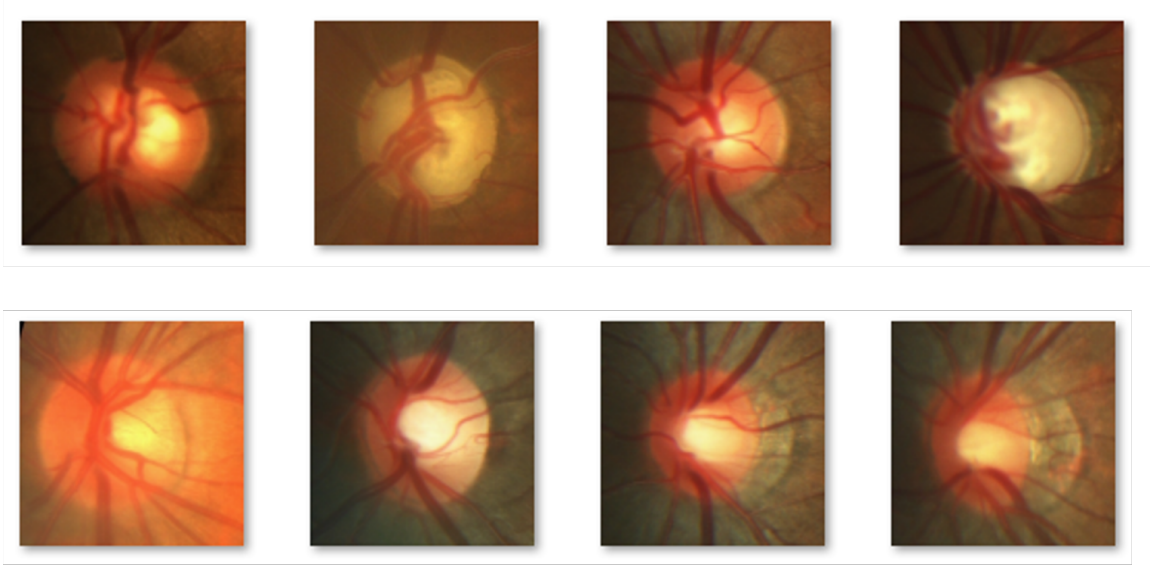


Figure 2.1: Sample images from REFUGE training set captured using ZEISS Visucam 500

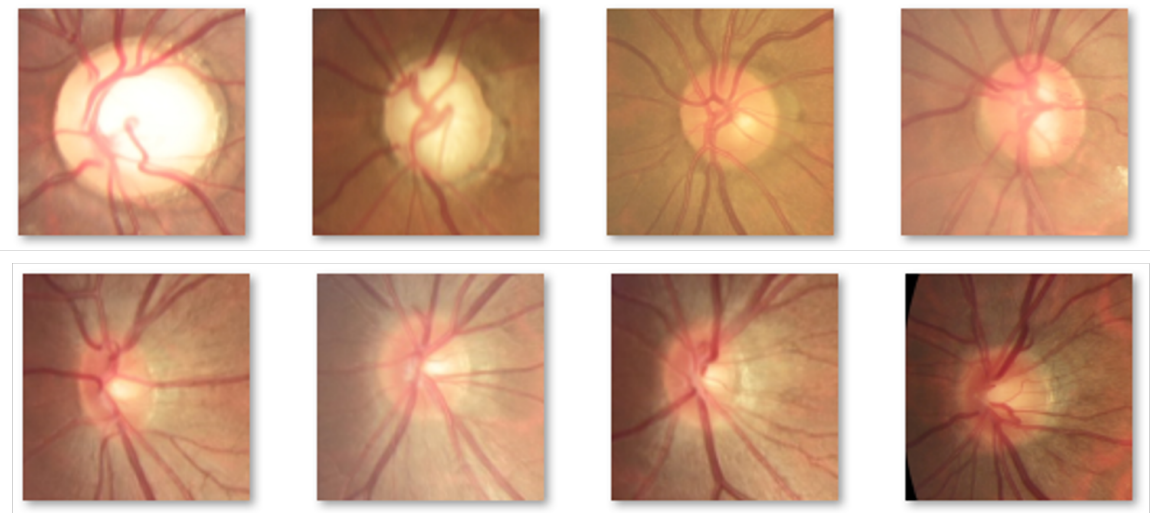


Figure 2.2: Sample images from REFUGE validation and test sets captured using Canon CR-2

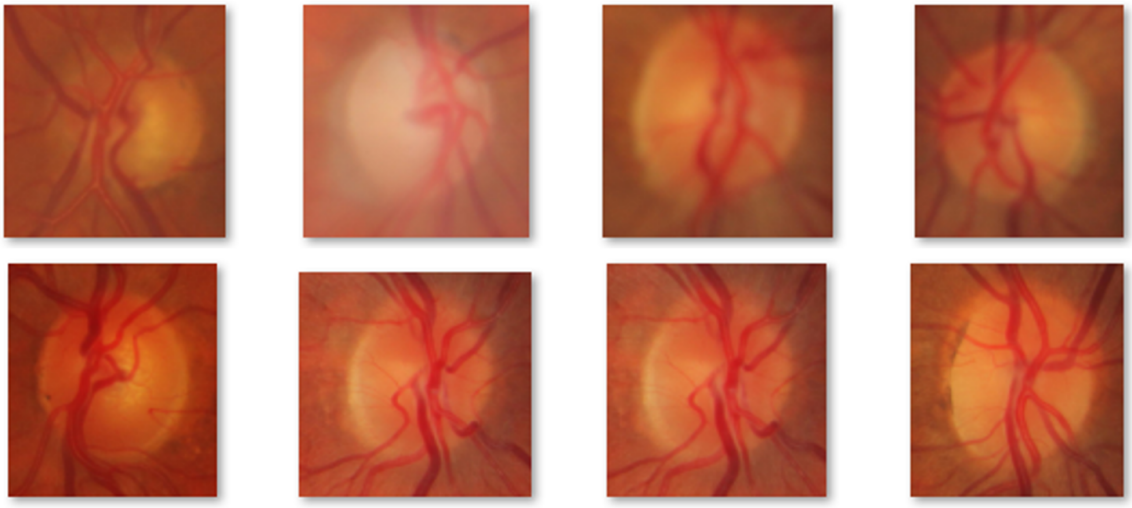


Figure 2.3: Sample images from RIMONEr2 dataset captured using NIDEK AFC-210

the Optic nerve head (ONH).

2.5 Problem of camera dependency

We have conducted a background study to examine the problem caused by camera dependency. In this study, we evaluate the influence of the camera model on automated glaucoma detection algorithm accuracy.

ACRIMA and REFUGE, labelled datasets that have captured device information, were included in the analysis. REFUGE images, which are whole fundus images, were cropped around the ONH to align with the ACRIMA images, where the ONH is centrally located. Also, images were resized to 224px x 224px prior to the initiation of feature extraction, ensuring consistency in input size across the datasets. Features were extracted from images using the ResNet101V2 pre-trained neural network and processed using a random forest classifier for glaucoma classification. The experiment was conducted multiple times by assigning images from one camera as the training set and images from another as the test set. Furthermore, we kept the classification accuracy for images from a single device as a baseline.

We analysed 1905 images from 03 camera models to obtain the area under the receiver operating characteristic curve (AUROC) for the test set. When using images from two cameras in the experiment, the highest AUROC of 0.81 was given when the training set was REFUGE01's test and validation sets, and the test set was its training set. The lowest AUROC was 0.23 when the model was trained on a REFUGE training set and tested on ACRIMA images. The highest of all, 0.96 AUROC, was given when training and test data came from the same camera.

This background study indicates that the availability of a wide variety of camera models impacts the reliable automated detection of glaucoma from fundus images. However, accurate automated detection of glaucoma from fundus images is possible, but we should be careful to specify the cameras for which the model is compatible. The results of this study were published under the title of "Automated detection of glaucoma from retinal fundus images using a variety of fundus cameras" [32].

2.6 Causes of camera dependency in fundus photography

Chen et al. performed a quantitative comparison of fundus images by Optos P200DTx (Optos PLC) and Zeiss Clarus 500 (Carl Zeiss Meditec AG) Ultra-Widefield (UWF) fundus cameras to compare the relative number of retinal pixels and retinal area imaged. They conducted a single-centre retrospective cross-sectional analysis using 78 eye images of 46 patients. Among the two devices, the Optos P200DTx captured statistically significantly more retinal area in all four quadrants than the Zeiss Clarus 500 under no statistically significant difference in patient or technician preference or image acquisition time between devices[33].

Jili Chen compares images from three UWF fundus cameras (Topcon TRC-NW300, Canon CX-1 and Optos Daytona plus) against Zeiss Clarus 500 to

evaluate the performance. Nonmydriatic fundus photographs of 17 patients were collected from each device, and three independent retinal imaging experts evaluated them. They included six qualitative parameters: (1) retinal colour reproduction; (2) image clarity; (3) field of view; (4) penetration of opacity; (5) small pupil imaging (vignetting); (6) operator ease of use and patient comfort. A consensus grading was performed after an inter-grader reliability assessment to rank the fundus cameras based on their scoring. Results indicated that the camera ranking varied depending on the parameter evaluated. For (1), Topcon, Canon and Zeiss were superior to Optos. As for (2), the posterior pole vessels were best appreciable on Zeiss and Canon, and mid-peripheral and peripheral blood vessels were significantly clearer on Zeiss than on Optos. However, Optos and Zeiss UWF cameras were superior to Topcon and Canon on (3). For (4), Zeiss was better than Optos. In case (5), Topcon, Canon and Optos were disturbed by vignetting, but not Zeiss. As of (6), both Opots and Zeiss were ranked as easy to use, whereas Zeiss was more comfortable for patients [34].

Chapter 3

Literature review

This chapter thoroughly studies recent research in machine learning-based glaucoma classification using retinal fundus images.

3.1 Machine learning-based glaucoma classification

Numerous machine learning-based systems have been developed using retinal fundus images for glaucoma classification. These systems aid doctors with clinical decisions related to glaucoma diagnosis that lead to early or better treatment. This section describes studies that successfully used feature extraction-based machine learning, deep learning, transfer learning, and ensembles to classify glaucoma using retinal fundus images.

Figure 3.1 shows the summary of different glaucoma diagnosis frameworks. The framework (a) shows the classical machine learning-based fundus image classification method, which involves manual feature extraction followed by classifier construction. The second framework shown in (b) uses a deep learning pipeline that classifies an image into a glaucoma class. Framework (c) uses transfer learning-based glaucoma detection. It first trains a model using a source dataset and then fine-tunes it for another dataset.

We studied related research work from 2018-2022, which was listed on

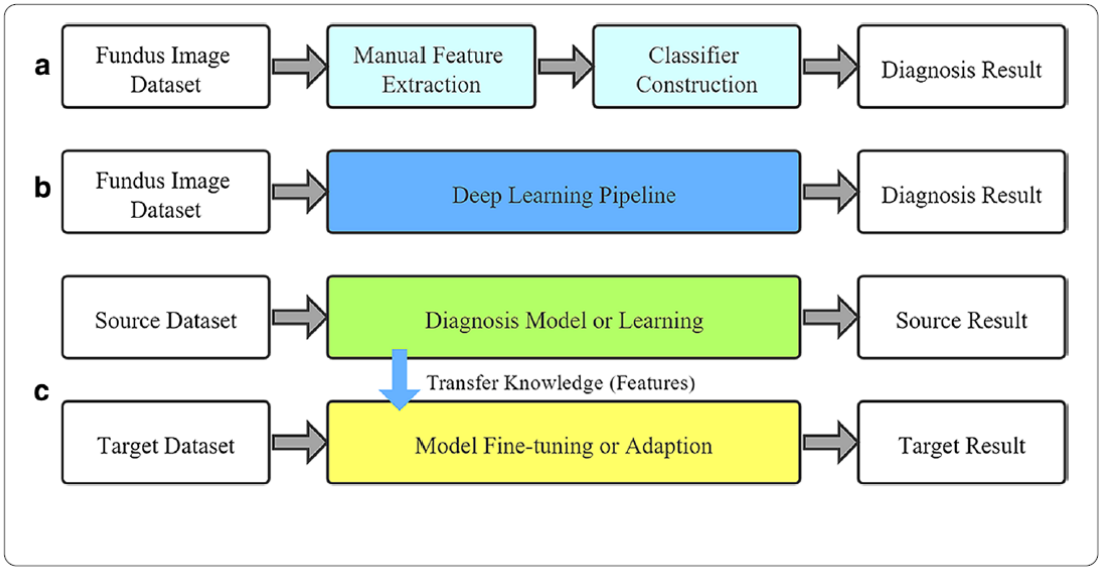


Figure 3.1: Different automatic glaucoma detection frameworks.

the Google Scholar platform. Approximately five publications per year were selected to include around 30 papers. "machine learning-based glaucoma classification" was the search term. An automated tool was employed to scrape research papers from the Google Scholar platform, yielding 164 results. As illustrated in Figure 3.2, we excluded inaccessible papers, removed duplicates, and omitted review articles to focus solely on original research. Furthermore, based on the titles, we excluded a majority of publications that did not pertain to classification systems. Finally, articles addressing other ocular diseases or employing different cameras were also excluded.

Apart from binary glaucoma classification systems, there are systems for glaucoma stage classification and systems that classify multiple eye diseases from fundus images. However, the latter two types of systems introduce complexity to the problem domain we explore because of the difficulty of finding publicly available labelled datasets belonging to multiple cameras. Hence, we excluded such papers and considered only the systems with binary classification for glaucoma and non-glaucoma/normal.

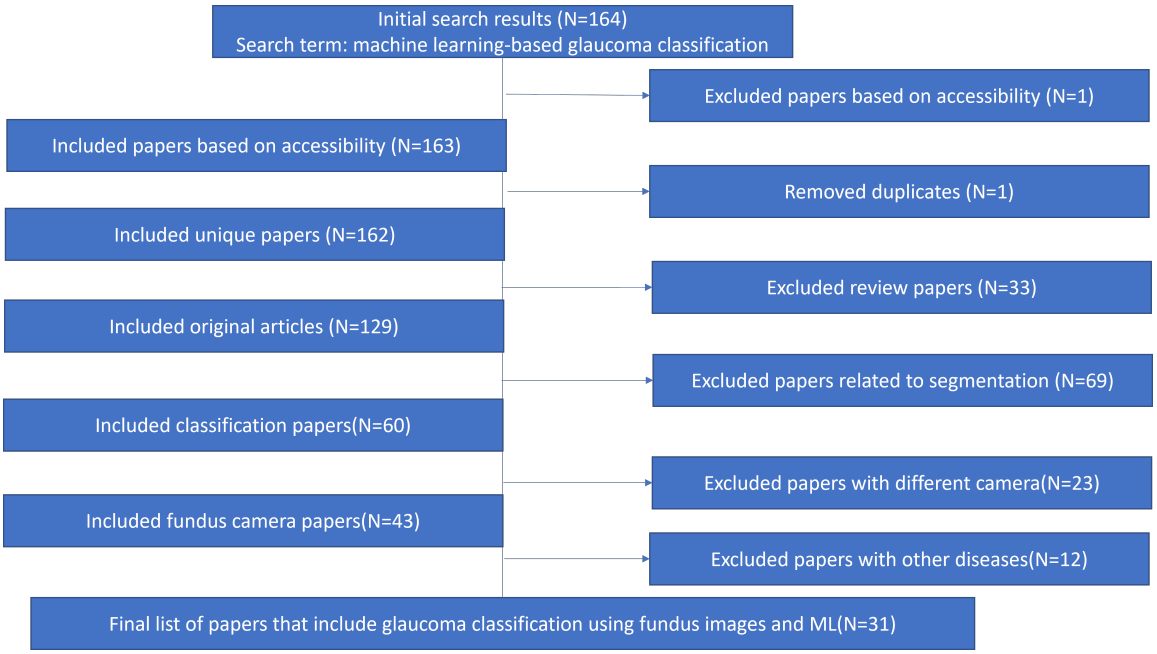


Figure 3.2: Systematic literature review.

3.1.1 Heuristic methods

Among the methods of feature extraction for glaucoma classification, there are segmentation-based methods, mathematical methods, and transfer learning-based methods for extracting various types of features, such as visual features and medical features/parameters. In these systems, clustering and classification methods have been used in the segmentation/ feature extraction and diagnosis phases, respectively.

Khan et al. propose a glaucoma classification model that extracts eight statistical parameters: contrast, energy, mean, homogeneity, entropy, standard deviation, variance and root mean square for each RGB plane, having 24 features in total. They perform image preprocessing using discrete wavelets transform-based bivariate shrinkage method before feature extraction. Following, features are selected using neighbourhood component analysis. The final feature set is fed into a least square support vector machine classifier and shows maximum classification accuracy of 91.22% for the RIMONEr2 dataset [35]. A system for glaucoma diagnosis in fundus eye images using diversity indexes

was developed by Araújo et al. Texture descriptors in the optic disc region are represented through diversity indexes. They extract 120 features from RGB channels and the grey channel, 30 from each. Then, a feature vector is undergone through a genetic algorithm to select the most significant features to be classified by a support vector machine (SVM). Tested on RIMONEr2, the classifier reaches an accuracy of 93.41%, sensitivity of 92.83% and specificity of 93.69% [36].

A real-time glaucoma classification system that uses object detection for feature extraction was proposed by Thanh et al. in [37]. The darknet YOLOv3 object detector was used to extract the optic disc and cup. They calculate medical parameters/features from disk and cup information. One thousand six hundred seventy-seven images were used to train the model and validated on 500 images. The accuracy, sensitivity, specificity, and precision are 92.0%, 92.68%, 91.34% and 91.2%, respectively [37].

Authors of the paper "An Efficient Deep Learning Approach to Automatic Glaucoma Detection Using Optic Disc and Optic Cup Localization" used an EfficientNet-B0 feature extractor to extract the deep features from the glaucoma suspicious images. Next, a bi-directional feature pyramid network performs a top-down and bottom-up key-point fusion several times to localise an area containing the glaucoma lesion. Finally, they train a deep learning model using the key points and predict the class. Moreover, they performed a cross-dataset validation using three public datasets (ORIGA, RIMONE DL and HRF), where they trained the system on one dataset and tested using the remaining datasets. The average test accuracy is between 97.9% - 98.9% [38].

Sudhan et al. in [39] introduces a glaucoma identification system that uses DenseNet-201 for feature extraction and a deep convolution neural network for classification. The model was tested on the ORIGA, a publicly available labelled dataset. First, they segment images using a U-Net to extract OD and optic cup (OC). The segmented image is then forwarded through a DenseNet-201 followed by a three-layer deep convolution neural network. The results

show an accuracy of 96.90% on 30% of ORIGA images [39].

Kumar and Kumar developed a system that uses an SVM for the early detection of glaucoma by providing CDR and parapapillary atrophy (PPA) as diagnostic features extracted from fundus images. OD contains a vascular tree, and the authors used its horizontal and vertical edges to extract ROI that includes CDR and PPA. They used K-means clustering and polar transformation to calculate CDR. PPA, an early sign of glaucoma, was extracted from the image using an ellipse's polar transform and direct least-square fitting algorithm. A k-means clustering algorithm was further used for the optimal separation of PPA from the residue area. An SVM was trained with 5-fold cross-validation using 141 images of glaucoma and healthy. The authors report accuracy, sensitivity and specificity of 97.3%, 100% and 88.2% for 39 test images, respectively [40].

Elangovan and the team used a pre-trained DenseNet201 model to extract features from the images of the DRISHTI-GS1 dataset. Then they classify images using SVM, Naive Bayes, K-nearest neighbour (KNN) and softmax to compare them. Accuracy, sensitivity and specificity for each of them are SVM-96.34%, 98.34%, 92.70%; Naive Bayes-95.56% 96.64% 93.30%; KNN-93.88%, 97%, 88.20%; and softmax-96.48%,98.88%,92.10%. The results indicate that softmax outperforms other classifiers [41].

Lima et al. diagnose glaucoma over retinal fundus image through deep features extracted using pre-trained networks as feature extractors [42]. They extract features from the RIMONE dataset [43] using five pre-trained networks. Random forest and logistic regression were used as classifiers. The logistic regression method achieved the best AUCs for RIMONEr2 and RIMONEr3 datasets with values of 0.957 and 0.860 when features were extracted using ResNet [44] and InceptionResNet [45] respectively.

Nayak et al. introduce an evolutionary convolutional network for automated glaucoma detection using fundus images named as ECNet. The network extracts discriminative features using four layers: convolutional, com-

pression, rectified linear unit (ReLU), and summation layer. Feature vectors are produced by an optimised network using an evolutionary algorithm called a real-coded genetic algorithm. Flattened feature vectors are then forwarded through different classifiers such as SVM, KNN, backpropagation neural network (BPNN), extreme learning machine (ELM), and kernel ELM (K-ELM) to select the best model. They tested the system using a private dataset and the ECNet model with SVM, resulting in 97.20% accuracy, the highest [46].

3.1.2 Transfer learning/deep classifier based methods

Many of the latest glaucoma classification systems use transfer learning or deep learning models. They are more popular because a single network performs feature extraction and classification. However, these methods require heavy training and parameter tuning and expect very large datasets. Also, preprocessing methods can be applied before training, and augmentations can be made to enlarge the datasets. In this subsection, we study those types of recent research work found in the literature.

Hirota and the team used ImageNet pre-trained VGG16 model to fine-tune for glaucoma classification. Their method was based on training separate models for each colour channel of RGB of an image. The AUROC of the RGB model was 0.800; the red model had an AUROC of 0.746, the blue model achieved only 0.558 AUROC, while the highest AUROC of 0.946 was given by the green model [47]. Another system proposed by Yedukrishnan et al. utilises the same architecture trained on the ACRIMA dataset, producing training accuracy of 94.4% and validation accuracy of 98% for 50 epochs [48].

The ResNet architecture was used by Borwankar et al. to classify combined data from DRISHTI [49] and REFUGEE [30] datasets. Vertical and horizontal flipping, rotation with an increase of 30, and translation within the range of 0.2 in both horizontal and vertical directions were used as augmentation techniques to achieve a nearly balanced dataset. They used a compact version of ResNet that attains an accuracy of 98.9% and an F1 score of 98.8% [50].

Asaoka et al. has conducted an extensive study to validate a pre-trained generic ResNet model for glaucoma screening using images from different fundus cameras. Training images were captured from the NONMYD WX camera (Kowa Company, Ltd., Aichi, Japan). They used images from two other cameras, NONMYD 7 camera (Kowa Company, Ltd.) and RC-50DX (Topcon Co.Ltd.), to test the system separately. All the data are privately collected in three different clinical settings, and the images of the control class are free from other pathologies. Additionally, with computationally expensive preprocessing methods and data augmentation, they reported over 99% testing AUROC[51].

Shoukat et al. uses EfficientNetB7 architecture to automatically detect glaucoma in its early stages from retinal fundus images. They apply a median filter, Gabor filter and adaptive histogram equalisation as preprocessing methods. Scaling, rotating and flipping were used to generate multiple images from a single image to enlarge the dataset. The system is tested independently using DRISHTI-GS and G1020 datasets. Best results were produced with an accuracy of 98%, a sensitivity of 95.19% and a specificity of 94% on the DRISHTI-GS dataset [52]. The authors expand the research by testing the same architecture by applying CLAHE followed by a median filter as preprocessing method and adding REFUGE as the third dataset to test the model. The best accuracy of 99.2% was given for the G1020 dataset with a sensitivity of 98% and specificity of 97% [53].

Retinal image analysis for glaucoma detection, developed by Sharmila and Shanthi, uses an ImageNet pre-trained InceptionV3 model for transfer learning. Experimented with the ORIGA dataset, the model achieved a test accuracy of 91.36% in a minimum of 20 numbers of epochs [54].

Manop Phankokkruad evaluates four deep transfer learning models in glaucoma detection for clinical application. Resizing and augmentations were done prior to training. Four hundred images from the REFUGE dataset were used to train and test the models. Results illustrate that each model VGG16, Xception, InceptionV3, and ResNet50V2 can achieve the accuracy of of 87.24%,

89.32%, 88.03%, and 76.52%, respectively [55].

Serte and Serener performed extensive experiments in developing a generalised deep-learning model for glaucoma detection. They test the model using five datasets by creating a training set combining four datasets and testing it on the remaining dataset. They report AUROC per each test set for ResNet-50, ResNet-152 and GoogLeNet models. Overall, there is an 80% comparability of results with previous work [56].

Saxena and the team used a convolutional neural network (CNN) to detect glaucoma from fundus images automatically. The proposed network has six layers comprised of four convolution layers followed by two fully connected layers. Before feeding into the network, the optic nerve is extracted using the ARGALI method. The model was evaluated using the ORIGA public dataset with AUROC of 0.822 [57].

Transfer-induced attention network introduced by Xu et al. uses transfer learning from the ophthalmic domain (DR). It is later used to extract discriminative features and finally classify for glaucoma identification using CNN. The network achieves an accuracy of 76.6%, a sensitivity of 75.3%, a specificity of 77.2%, and AUROC of 0.835 when tested with the ORIGA dataset [58].

Santos at. al. used a capsule network (CapsNet) [59] Santos for identification of glaucoma in retinal images. It consists of regular convolutional layers and two convolutional capsules, one for intermediate feature mapping and another for classification. The network was tested on a combined dataset that includes images from RIMONEr2 and Drishti-GS databases. The results were promising, with 90.90% accuracy and 0.904 AUROC [60].

The Glaucoma classification network produced by Juneja et al. was a CNN designed using 76 layers, including convolutional, pooling, fcn and a final output layer. They crop the fundus image around ONH augment and denoise before training. They use a combined dataset from images of RIMONEr2 and Drishti-GS and report an accuracy of 97.51%, sensitivity of 98.78% and specificity of 96.20% [1].

Elangovan and Nath did a glaucoma assessment from colour fundus images using CNN. An 18–layer CNN comprised of four convolutional layers, two max-pooling layers, and one fully connected layer was designed and trained to extract the discriminative features and classify the fundus image. The network was tested on DRISHTI–GS1, ORIGA, RIM–ONEr2, ACRIMA, and LAG databases. Images were rotated to increase the dataset and used the 7:3 ratio for training and testing data split. Accuracy of 86.62%, 78.32%, 85.97%, 96.64% and 94.43% for each of the respective datasets above [61].

3.1.3 Ensemble methods

Ensemble methods diagnose glaucoma based on the combined predictions of two or more independently implemented systems. Usually, they consist of feature extraction-based methods and classifier-based methods in a single system. The final prediction is made by majority voting, average voting, or assigning weights for each subsystem. This sub-section studies several recent works on ensemble methods.

Civit-Masot et al. developed a dual machine-learning system to aid glaucoma diagnosis using disc and cup feature extraction. The first subsystem applies two U-Nets to independently detect the optic disc and cup and calculate CDR based on physical and positional features. The second subsystem consists of a lightweight Imagenet pre-trained MobileNetV2-based classifier to classify complete eye fundus images. The voting system that decides a glaucoma patient based on any subsystem's positive prediction improves the final detection's sensitivity to 0.91 [62].

Prashanth et al. proposed an early and efficient glaucoma detection system using image processing and deep learning. The first subsystem was based on image processing that extracts the optic cup and disk using various image processing techniques. Afterwards, they calculate CDR to decide on the glaucoma class. The second subsystem used a custom CNN and VGG19 model to compare the results of deep models. Their results indicate that CNN yields better

accuracy than pre-trained VGG19, and they report 84.51% accuracy. The authors have developed a GUI that provides annotations and useful information to the system users [63].

The ensemble method developed by Deepa et al. comprised three deep learning models, ResNet, VGGNet and GoogLeNet, as feature extractors. They used CLAHE as the preprocessing technique, and rotation, horizontal and vertical flipping were used as augmentation techniques. Each ImageNet-trained network produces a feature vector. The final feature vector is computed using majority voting for feature selection. The Softmax classifier layer is used for the final classification for glaucoma identification. The proposed method was tested using five public glaucoma datasets independently and in combination. The best accuracy of 91.13% was given for the PSGIMSR dataset, while the combined achieved an accuracy of 88.96% [64].

Serener and Serte proposed a glaucoma classification system via deep-learning ensembles. They used GoogLeNet, ResNet-50, and ResNet-152 as single classifiers. Furthermore, they tested the pair-wise ensembles and all three together as well. The authors used the sum of the probabilities (SP), a product of the probabilities (PP), the sum of the maximal probabilities (SMP), and majority voting (MV) methods as fusion methods. Model training was performed by combining four of five publicly available glaucoma datasets as training set and tested on the remaining dataset selected from HRF and Drishti-GS1. Other three datasets are sjchoi86-HRF [65], RIMONE [43], and ACRIMA [66]. Based on the results, the authors conclude that ResNet-152 is the best when a single classifier is used. The results for combining two or three architectures are the same. Also, the majority voting fusion method shows a better classification accuracy [67].

3.2 Problems in existing machine learning based glaucoma classification methods

In all of the above work, many common aspects can be seen. We will discuss them under different topics, such as image data, preprocessing, augmentation, features, classifiers, network training and evaluation. Overall, there are several predominant limitations in the above studies. These limitations are identified as gaps in our study, and we try to remove the gap by mitigating or improving the scenarios.

3.2.1 Issues with respect to image data

In the existing literature, many approaches focus on analysing either a specific region of interest (ROI) within an image through segmentation techniques, or they examine entire retinal fundus images in their analyses. Previous studies generally adopted one of these methods. In our preliminary investigation, we employed both approaches by processing both the entire image and the ROI.

Most of the studies have tested their method against one dataset. All the training, validation and test set images are gathered using a single camera. Also, many of the methods we discussed above are tested and used with private datasets that question the external validity of such a method. Furthermore, the used datasets are relatively large, whereas we find smaller datasets in clinical environments in the medical domain. Some of the public datasets contain imbalanced data between classes. Usually, it is more difficult to find glaucomatous images than non-glaucomatous images when collecting data. Due to privacy and ethical considerations, the transfer of images within research centers or clinics is restricted. Hence, the dataset has fewer images in the class we are more interested in. It is important to note that labelling the data uses different methods such as considering multiple other tests, expert annotation and aided by annotation software. Moreover, some datasets include glaucoma suspects and glaucoma stages in the glaucoma class. All in all, there are dif-

ferent ways we can test a certain method using multiple datasets to validate usability.

One of the challenges in deploying deep learning models is their requirement for training on extensive datasets, typically consisting of hundreds of thousands of images. In contrast, our models were trained on a relatively small dataset comprising only a few hundred images. While adopting a generative AI approach to artificially augment our dataset was considered, our objective was to replicate the actual clinical setting, where typically few images are available for glaucoma, compared to a larger number of images for normal cases. Consequently, it was imperative to proceed without generating synthetic data. Therefore, we concentrated on adapting our algorithms to work effectively with the available data, maintaining the integrity of the original datasets without recourse to artificial augmentation. This approach was central to our methodology, aligning with the specific conditions and limitations of clinical environments.

3.2.2 Issues with respect to preprocessing methods

Image preprocessing is the steps taken to format images before using them in model training and inference. Image enhancements, denoising and normalisation, are commonly applied using image processing techniques [68, 66, 47, 61]. It includes but is not limited to resizing, orienting, and colour modifications. Afterwards, we can achieve an acceptable image to be fed into the network. The most common preprocessing method found in glaucoma classification is resizing which reduces the number of pixels in an image. It is important for further processing because fundus data are often high-resolution images.

Furthermore, most existing machine-learning models require a specific image size, such as 224 x 224 or 229 x 299. Hence, resizing has become an essential preprocessing step in machine learning-based classification. Image cropping is the second most common preprocessing method performed manually or automatically around the ONH of a retinal fundus image. It removes

redundant information from an image and limits it to an ROI. Many of the public datasets are centre cropped around ONH.

However, many other preprocessing techniques, such as image normalisation, histogram equalisation, and image filtering, have been applied. These are mostly found in systems that use feature extraction-based methods. The literature includes examples of using single or very few, i.e. two or three transformations at once. Existing systems typically apply preprocessing methods and augmentations randomly. The selection of operations and their parameters can be optimised for better performance. However, there are possibilities of applying multiple preprocessing methods at once as a combination.

3.2.3 Issues in image augmentation

Image augmentation is the manipulations applied to images to create different versions of similar content to disclose the model to a broader array of training examples. Examples include randomly altering an input image's rotation, brightness, or scale. Thus a model considers what an image subject looks like in various situations. Augmentation manipulations are forms of image preprocessing, but they are different: image augmentation is only applied to the training data, while image preprocessing steps are applied to training and test sets. Thus, a transformation used as an augmentation step in some situations may best be a preprocessing step in others.

Augmentation techniques are used in almost all the studies mentioned above. The primary purpose of data augmentation is to increase the number of images in a dataset before training. It applies an image processing technique to an image and adds the transformed image to the dataset. In other words, augmentation enlarges a dataset by generating new data based on the existing data. The most common augmentation techniques in glaucoma detection systems, including random rotation, horizontal flip, and vertical flip, are discussed by [48].

The literature includes examples of using single augmentations or very few,

i.e. two or three transformations at once. They are usually random and need to be optimised. One can apply augmentation online or offline. The offline method, also known as static augmentation, adds new images to the dataset before training, which can be stored for later use. Online or dynamic augmentation, on the other hand, augments data at the beginning of the training process when feeding the data into the machine learning model. Multiple augmentation techniques can be simultaneously applied as a combination. Furthermore, stacked transformations can serve as an augmentation method prior to training a machine learning model.

3.2.4 Issues in feature extraction

Features play an important role in machine learning-based glaucoma classification. There are medically important features such as CDR, which can be calculated using the extracted OD and OC information from a fundus image. Various methods have been suggested in the literature that utilise image processing and segmentation techniques, such as clustering and unsupervised learning. Other medically accepted features, such as retinal nerve fiber layer (RNFL) thickness, can be calculated by segmenting nerves in a fundus image [69]. However, since image segmentation is not the focus of our research, we will not discuss them further.

Another type of features extracted using fundus image feature extraction methods is statistical measures such as mean value, variance, standard deviation of pixels etc. These values are calculated per image and stored as a feature in a feature vector to be fed to a classifier.

Furthermore, spatial features are extracted to reduce the dimensions of data. Fundus images usually contain RGB colour information in high resolution. Texture, higher-order spectra, wavelet energy features, and pixel orientation and density are commonly used based on image features. Applying machine learning-based feature extraction can reduce the 2D feature space to a 1D feature vector. Pretrained deep learning models without the top layer

are commonly used as feature extractors.

In previous studies, systems for glaucoma classification were often designed to utilize either image-based or statistical features [11]. Nevertheless, there exists the potential to enhance classification accuracy by concatenating different types of features.

3.2.5 Issues with respect to classifiers

There are two types of classifiers in machine learning: shallow classifiers and deep classifiers, based on shallow neural networks or architectures and deep learning models, respectively. Shallow classifiers can be named SVM, Naive bias, random forest, XGBoost, etc., while ResNet, EfficientNet, and Inception are some deep learning models used as classifiers. Both types of classifiers have been employed in the literature to classify glaucoma using retinal fundus images.

From the studies listed in Section 3.1, [36, 40, 41, 42, 46] have employed shallow classifiers in their research. Conversely, [47, 48, 51, 52, 55] utilised deep classifiers in their studies. The former was used for testing the image processing and feature extraction techniques, while the latter was used for testing SOTA models for glaucoma classification using retinal fundus images. Shallow classifiers usually require image processing and/or feature extraction of images prior to classification, whereas deep classifiers include internal feature processing but require more processing resources.

3.2.6 Issues in network training and evaluation

Most of the recent work studied above used various network hyper-parameters such as learning rate, optimisers, metrics, number of epochs, batch size etc. Every study occupied the SOTA values or algorithms when selecting them, but most systems' epochs were below 100 (25, 30, 50). The graphs of accuracy and loss plotted against epochs show that they have yet to achieve stable results. One example can be found in Figure 3.3, which uses 30 epochs to train the

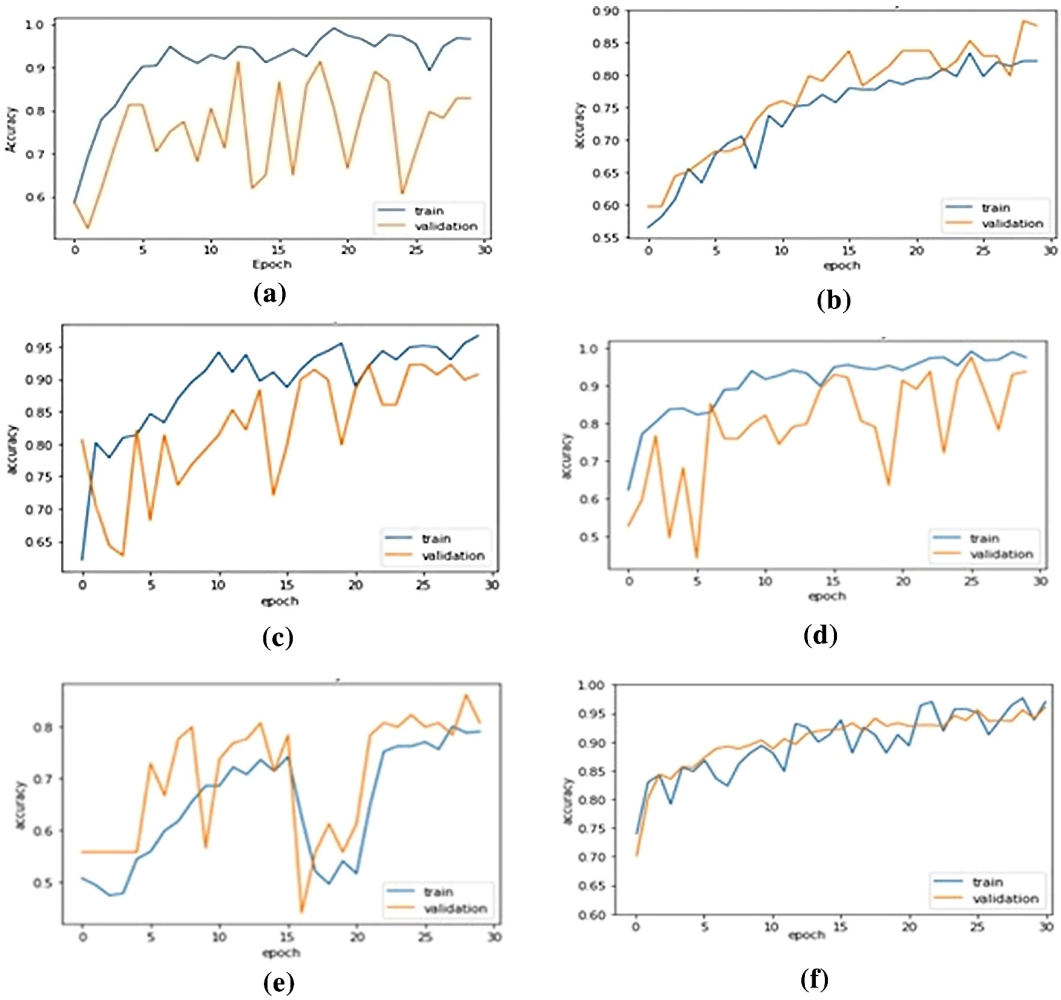


Figure 3.3: An example of unstable accuracy graphs [1]

network where the graphs are not converged or plateaued [1].

Also, most systems are trained with a single dataset and validated and tested using the same dataset by splitting it into three, as illustrated in Figure 3.4. Therefore, a system always sees similar images in all three subsets and, on average, results in higher training/validation/testing accuracy. It questions the re-usability of a system with images from another camera. However, a handful of experiments are done with single training and testing with another device, but they are done on private datasets. Hence, we can not reproduce the results and test the system in another environment. However, a few experiments are performed on multiple device-based datasets, but the models are trained on a combination of them and tested on a single device. By studying these systems,

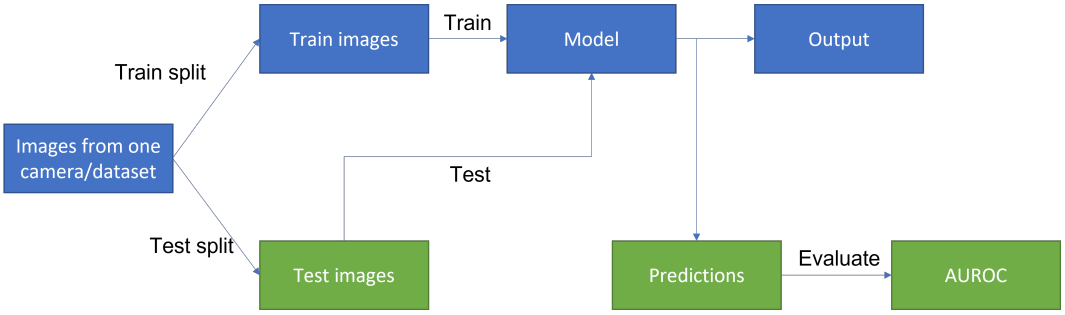


Figure 3.4: Standard method of train/validation/test splits using a single camera/dataset.

we design our experiments to tackle the unseen domain problem.

3.3 Summary of review

Considering the shortcomings and gaps identified above, we design our research and experiments to improve and mitigate the identified issues in Section 3.2.

First, we use images from various retinal fundus cameras widely used in clinical environments. We gathered publicly available labelled datasets that include camera information. Some datasets contain ten to 1000 images per class of glaucoma or non-glaucoma. Also, several datasets are highly imbalanced as discussed in the Section 3.2.1, while others are relatively balanced. Using those diverse and challenging datasets, we conducted multiple experiments to train machine learning classifiers and tested them across different cameras.

Additionally, the experiment setups were designed such that the unseen domain problem is addressed properly. For example, one of our experiments uses images from three devices, each as train, validation and test set during one training instance. Also, we shuffle the datasets and train the system for each device's data. Moreover, we used a combination of datasets in another experiment to see the effect of using images from multiple devices to train a system and test on a third device. Importantly, we compared the results of these experiments against the same device scenario, keeping it as a baseline.

These experiments were conducted to test the validity of preprocessing approaches that have never been applied to retinal fundus images in glaucoma classification systems together as a combination. We tested three preprocessing methods: median filtering, input standardisation and multi-image histogram matching. Importantly, the third method was introduced in our study, and it generates a transformed image similar to neural style transferring (NST). We used combinations of these methods across training and testing data. The experiments used pre-trained models as feature extractors of preprocessed images, and the features were used to train random forest and XGBoost classifiers. We compare the results to select the best ones.

Finally, we extend the experiments to use NST, one of the most recent methods that claim to work well with medical image segmentation tasks. The idea was to apply the technique in test time to augment the images to match the style of another dataset. We used publicly available pre-trained machine learning models. However, those are tested only with one dataset that it was trained on. We were interested in examining the performance of such a method tested on multiple publicly available datasets derived in different clinical environments.

Chapter 4

Comparing pre-trained neural networks as feature extractors

This chapter explains the initial study conducted to compare multiple pre-trained deep neural networks as feature extractors. We aimed to identify the best feature extractor(s) and the best feature sets to train a shallow classifier to experiment with the REFUGE training set. Testing the final system with test data from the same dataset captured by a second camera reduced the classification accuracy, establishing the background for our entire study.

4.1 Pretrained networks as feature extractors

In image processing, pattern recognition and machine learning, features are identified as derived values built from a set of measured data. The features are non-redundant but informative data representations that help future learning and generalisation. Feature extraction and dimensionality reduction have a strong relationship, as feature extraction usually results in reduced dimensions [70]. Sometimes, algorithms require more time and memory to process large input data. We can reduce features by removing redundant features, such as repeated pixel values in an image and converting the input data into a feature vector using feature extraction. As a result, the expected task can be performed using the new representation instead of the initial complete data

[71].

Feature extraction is a widely used technique in machine learning. It is extremely helpful when working with high-resolution images. Retinal fundus images are examples of high-resolution images with 1000-2500 pixels on each edge of the picture. Pretrained machine learning models can be used as feature extractors to get feature vectors having 512-4096 features.

Keras API [72] is a Python-based machine learning library that includes 26 pre-trained networks on the ImageNet dataset. The dataset has images belonging to 1000 classes, considered benchmark a dataset in machine learning classification. Publicly available weights of those pre-trained models can be used to pass images through the model and generate features by removing the top layer of the network. We use these features to train two classifiers and predict glaucoma possibility in cross-validation experiments.

4.2 Classifiers

4.2.1 Logistic Regression

Logistic regression is a statistical model used in machine learning and most medical field research. This classifier can model the probability of a certain class against features in binary classification. Ahn et al. in [68] first used logistic classification with fundus photographs collected from a clinical setting using a non-mydriatic auto fundus camera (AFC-330, Nidek, Japan) with a total of 1,542 photos. They cropped the photos at the optic nerve region with a size of 240x240 pixels. Images were flattened to get a one-dimensional array before inputting into the classifier. The model's training accuracy was 82.9%, and the AUROC was 0.85 [68]. However, they have not performed feature extraction prior to classification. Since they used a private dataset, using their results as a baseline or reproducing them is infeasible.

4.2.2 Random Forest

Random forest is an ensemble learning-based classification method. It builds an ensemble of decision trees at the training time and predicts the class by combining the predictions of individual trees [73]. It computes the average of many decision trees through bagging. Here, samples are chosen with replacement (bootstrapping) and combined (aggregated) by taking their average. Bagging is the shortened word for "bootstrap aggregation". Random forests require very little hyperparameter optimisation yet perform at near SOTA levels for many problems.

A recent study has used random forest classifier-based techniques for glaucoma detection, as stated in the previous chapter [42]. Another research by Acharya et al. [12] in 2011 used a random forest classifier for glaucoma identification. They used ten trees with unlimited tree depth. Each tree "votes" for one of the two classes and the most popular class is assigned. However, they used a much smaller private dataset of 60 images to test the system for texture and higher-order spectral features after z-score normalisation and feature selection and reported an accuracy greater than 91%.

4.3 Experiments

We designed an experiment to test 26[1] pre-trained machine learning models as feature extractors for glaucoma image classification using two conventional classifiers. The initial study aimed to compare different models and identify the best feature extractor, best classifier and best feature representation. We experimented with three steps: (1) data acquisition, (2) feature extraction and (3) ten-fold cross-validation.

[1]There were only 26 models listed in `https://keras.io/api/applications/` when this study was initiated in 2020

4.3.1 Tools

The experiments were developed and executed in a server with an installed Ubuntu 16.04.3 LTS operating system. We created a virtual environment that uses Python 3.5. The libraries Keras API version 2.31 and scikit-learn version 0.22 were used as the main software packages to utilise feature extractors and classifiers, respectively.

Keras is a high-level neural networks API written in Python capable of running on top of TensorFlow. It was developed with a focus on enabling fast experimentation. Keras has deep learning libraries that allow easy and fast prototyping, support convolutional networks and run seamlessly on CPU and GPU. This API is user-friendly, modular, easily extensible and works with Python.

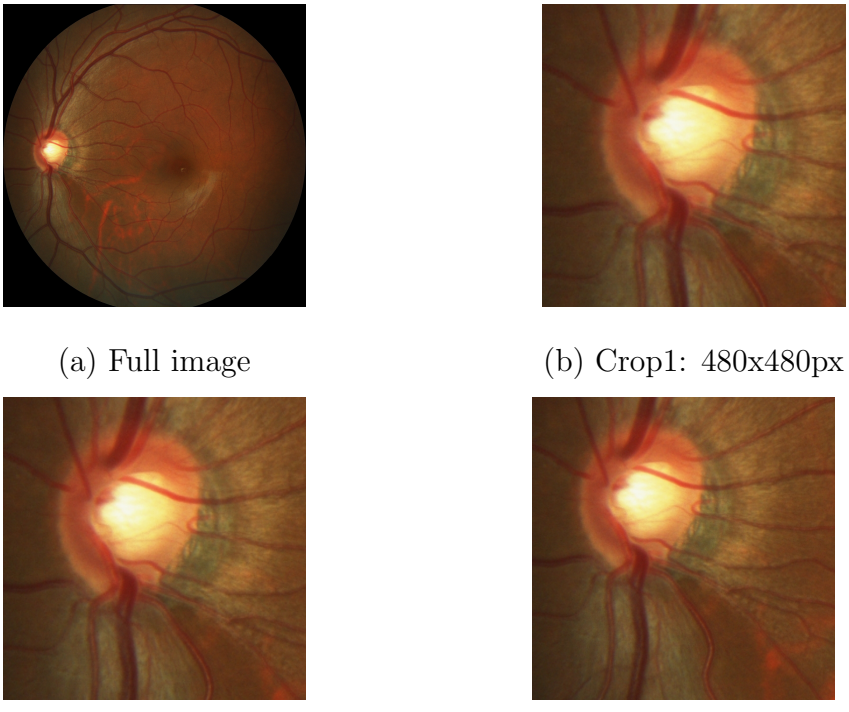
Scikit-learn (*sklearn*), on the other hand, is a Python-based free and open-source machine learning library that includes many clustering, classification and regression algorithms such as k-means, DBSCAN, gradient boosting, logistic regression, support vector machines and random forests. It uses numerical and scientific libraries such as NumPy and SciPy in Python.

4.3.2 Dataset

We used REFUGE01 [30] dataset to experiment with our proposed method. A detailed description of the dataset is given in Subsection 2.4.1 under Section 2.4 in Chapter 2. The dataset includes whole retinal fundus images in three sets, train, validation, and test, each having images belonging to two classes, glaucoma and nonGlaucoma. In our experiments, we used the same ground truth labels (glaucoma/healthy) as specified by the reference standard for Task 1 in the REFUGE challenge [30]. When performing the cross-validation experiment, we rearranged the images by combining images of all three sets and splitting them into two classes.

4.3.3 Experimental setup

We executed multiple tests using whole and cropped images. We applied the template matching technique to crop around the ONH to prepare the cropped version of the images. The first image of the training set was manually cropped to locate the template for ONH. The remaining images of the dataset were automatically cropped using the OpenCV version 4.3 [74] with the normalised correlation coefficient method in the template matching algorithm. Each crop image had three different cropping sizes, 480, 520 and 600 pixels vertically and horizontally, and the ONH was placed at the top left corner of the cropped image. This setting allows to capture RNFL defects in more detail than cropping a square centered in the ONH [30]. To maintain the accuracy of the dataset, I conducted a manual inspection of all cropped images to verify the correct inclusion of the ONH. Figure 5.2 shows sample images of the four prepared datasets.



(a) Full image (b) Crop1: 480x480px

(c) Crop2: 520x520px (d) Crop3: 600x600px

Figure 4.1: Sample images of dataset: raw image and cropped images around optic nerve head.

Secondly, we extracted features using 26 ImageNet [75] pre-trained deep neural networks. A max-pooling layer replaced each model's 1000-class classification layer (top layer). Features were extracted for the original and three cropped images, and the feature vectors were stored separately. The saved features of each cropped size were concatenated with the original features to create expanded feature sets.

Lastly, we conducted ten-fold cross-validation experiments using original and cropped image features (by themselves) and concatenated features (original plus cropped, of one size only). There were 400 images in the REFUGE1 dataset creating 40 images per fold. The base classifier was either one of logistic regression or random forest models. Since the REFUGE1 dataset is imbalanced, examples were weighted inversely according to label frequency to balance the label distribution.

We composed the random forest model using 1000 estimators and ten maximum features. The logistic regression model used l2 regularisation and liblinear as the solver. The model was trained for 1000 iterations. Finally, we obtained cross-validation prediction of the glaucoma class as an AUROC score from each model to assess the overall performance of the classification algorithm in a setting when a low number of false positives is tolerated [30]. Figure 4.2 illustrates an overview of the process.

4.4 Results and discussion

Our experiment results are summarised in Table 4.1 and Table 4.2, presenting data to four significant figures, consistent with the standards set by the REFUGE challenge [30]. There, feature type `Full` refers to the features extracted from original uncropped fundus images. `Crop1`, `Crop2` and `Crop3` represent crop sizes 480×480px, 520×520px and 600×600px respectively. Features generated after the concatenation of full features with each previous crop size are denoted by `Concat1`, `Concat2` and `Concat3`. Results only for the best

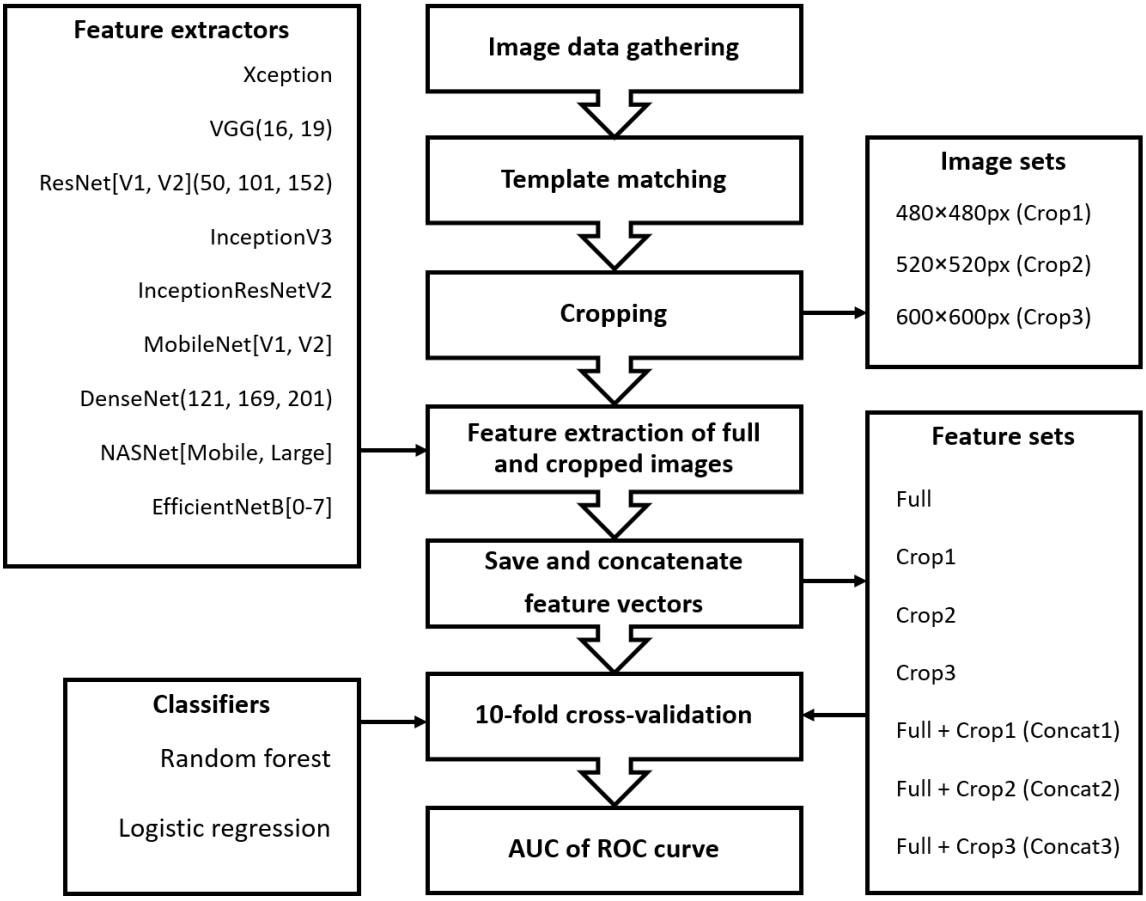


Figure 4.2: Experimental setup: 26 feature extractors × 7 feature sets × 2 classifiers = 364 cross validation experiments in total.

feature extractor for each feature type are given out of all 364 cross-validation experiments we performed.

Each feature extractor generated a certain number of features for full and cropped images, and the concatenation has doubled this number. The random forest model gave the best result of 0.9785 AUROC score when full features were combined with the `Crop1` image features extracted from ResNet101-V2. However, the model shows an AUROC of ≈0.93 for each of the `Concat2` and `Concat3` feature types as illustrated in the Tables.

On the other hand, the logistic regression model obtained the best AUROC score of 0.9441 for full features combined with the `Crop1` image features extracted from ResNet50. Furthermore, it achieves an AUROC score of ≈0.94 for `Concat2` and `Concat3` features when they are extracted using DenseNet169.

We compare all 26 feature extractors in Table 4.3 to show the median AUROC value over each feature set or feature set combination. It can be inferred that most feature extractors perform quite poorly compared to the best feature extractor. However, both classifiers achieve the best results for feature type `Concat1`.

All in all, features from the full feature set concatenated with the 480×480px cropped feature set (`Concat1`) are the best configuration. It increases the AUROC score dramatically, compared with the scores returned for individual crop image size or full image features alone by a random forest classifier. Once the features are extracted, the random forest classifier is better suited as a base classifier of retinal fundus images than logistic regression.

Table 4.1: Results for the best feature extractor for each feature type of Random Forest ten-fold cross-validation

Feature type	Feature extractor	Number of features	AUROC
Full	VGG19	512	0.9028±0.1312
Crop1	ResNet50V2	2048	0.9330±0.0953
Crop2	ResNet50V2	2048	0.9370±0.1906
Crop3	ResNet50V2	2048	0.9330±0.2859
Concat1	**ResNet101V2**	**4096**	**0.9785±0.3812**
Concat2	ResNet50V2	4096	0.9327±0.0615
Concat3	ResNet50V2	4096	0.9329±0.0615

Table 4.2: Results for the best feature extractor for each feature type of Logistic Regression ten-fold cross-validation

Feature type	Feature extractor	Number of features	AUROC
Full	ResNet101	2048	0.9014±0.1172
Crop1	DenseNet169	1664	0.9308±0.1841
Crop2	ResNet152V2	2048	0.9214±0.1199
Crop3	ResNet101V2	2048	0.9045±0.0667
Concat1	**ResNet50**	**4096**	**0.9441±0.0558**
Concat2	DenseNet169	3328	0.9417±0.0917
Concat3	DenseNet169	3328	0.9398±0.1025

Table 4.3: Median AUROC values given by each classifier for 26 feature extractors

Feature type	Logistic regression	Random forest
Full	0.7969±0.0994	0.8487±0.1237
Crop1	0.8203±0.1488	0.8688±0.0559
Crop2	0.8060±0.1154	0.8757±0.0707
Crop3	0.8202±0.0753	0.8763±0.0805
Concat1	**0.8539±0.1209**	**0.9285±0.0683**
Concat2	0.8454±0.0957	0.8978±0.0795
Concat3	0.8461±0.0925	0.8918±0.0936

We took a step forward and tested our method on the REFUGE2 validation dataset. Glaucoma labelling information for the dataset is publicly unavailable, but online testing can be done using the REFUGE Challenge platform. However, the model's performance in the second data set was poor,

and the images were captured using a different camera whose camera model information was hidden from the public. This conveys to take caution if test images are from a different or unknown source and highlights the need to use models primarily with images from cameras for which they have been trained. Hence, in this background study, developing models robust to change in camera is a potential topic for future research. The outputs of the initial study were recently published with the title of "Comparison of pre-trained feature extractors for glaucoma detection" [76].

4.5 Extended experiment

We extended the above experiment to extract features using pre-trained models "AlexNet" [77] and "MobileNetV3" [78] and to compare the results with previous experiments. Our experiment results are summarised in Table 4.4.

The logistic regression model gave the best result of 0.9811 AUROC score when full features were combined with the `Crop3` image features extracted from AlexNet. However, the model shows an AUROC of ≈0.95 for each of the `Concat1` and `Concat2` feature types as illustrated in the Table. On the other hand, the random forest model obtained the best AUROC score of 0.9721 for full features combined with the `Crop1` image features extracted from AlexNet. Furthermore, it achieves an AUROC score of ≈0.95 for `Concat2` and `Concat3` features when they are extracted using AlexNet.

Updated tables 4.5 and 4.6 compare all 28 feature extractors to show the best AUROC values over each feature set or feature set combination. It can be inferred that the AlexNet feature extractor shows better AUROC for most feature sets compared to the other feature extractors, including MobileNetV3. Furthermore, the random forest classifier maintains the best results for feature type `Concat1`. This enhanced performance is likely attributable to the absence of redundant features in `Concat1`, which are present in the other two feature types.

The computational process utilised 6 GB of memory. The feature extraction phase required approximately 15 minutes to generate a single feature set from the REFUGE1 dataset. Subsequently, feature concatenation was completed in an average time of 0.77 minutes. For the cross-validation, processing one feature set through two function evaluations in the REFUGE1 dataset took approximately 9.62 minutes.

Table 4.4: AUROC values given by each classifier for MMobileNetV3 and AlexNet feature extractors

Feature Type	Feature Extractor: MobileNetV3		Feature Extractor: AlexNet	
	LogisticRegressionCV	RandomForestClassifier	LogisticRegressionCV	RandomForestClassifier
Full	0.8406±0.1818	0.5311±0.1254	0.8809±0.0445	0.8888±0.1172
Crop1	0.8505±0.1063	0.5342±0.1247	0.9403±0.0736	0.9448±0.0105
Crop2	0.8542±0.2158	0.5090±0.1128	0.9045±0.0772	0.9364±0.1353
Crop3	0.8731±0.1680	0.5271±0.1375	0.9141±0.0676	0.9372±0.1244
Concat1	0.9206±0.0605	0.5185±0.1254	0.9608±0.0372	0.9721±0.0744
Concat2	0.9364±0.1606	0.5066±0.1499	0.9497±0.0563	0.9494±0.1223
Concat3	0.9594±0.1106	0.5244±0.1293	0.9811±0.0578	0.9592±0.0992

Table 4.5: Updated results for the best feature extractor for each feature type of Random Forest ten-fold cross-validation

Feature type	Feature extractor	Number of features	AUROC
Full	VGG19	512	0.9028±0.1312
Crop1	AlexNet	4096	0.9448±0.0105
Crop2	ResNet50V2	2048	0.9370±0.1906
Crop3	AlexNet	4096	0.9372±0.1244
Concat1	**ResNet101V2**	**2048**	**0.9785±0.3812**
Concat2	AlexNet	8192	0.9594±0.1223
Concat3	AlexNet	8192	0.9592±0.0992

Table 4.6: Updated results for the best feature extractor for each feature type of Logistic Regression ten-fold cross-validation

Feature type	Feature extractor	Number of features	AUROC
Full	ResNet101	2048	0.9014±0.1172
Crop1	AlexNet	4096	0.9403±0.0736
Crop2	ResNet152V2	2048	0.9214±0.0144
Crop3	AlexNet	4096	0.9141±0.1199
Concat1	AlexNet	8192	0.9608±0.0372
Concat2	AlexNet	8192	0.9497±0.0563
Concat3	**AlexNet**	**8192**	**0.9811±0.0578**

Chapter 5

Comparison of preprocessing methods against unseen domain problem

Our study evaluates the effect of domain generalisation using three image preprocessing methods over the accuracy of glaucoma detection algorithms concerning different fundus camera devices. Images from three distinct fundas cameras from different manufacturers, namely Canon CR-2, NIDEK AFC-210 and ZEISS VISUCAM-500, are used in our work. Sample images from three cameras are shown in Fig 5.1.

The objective of this experiment is to compare the combinations of multiple existing preprocessing methods in order to examine how well they improve device domain adaptation for the task of glaucoma detection in retinal fundus images. Random forest classifiers achieve the best accuracy among other base classifiers with these types of images and image features in a previous study [76]. Hence, we use the random forest as the base classifier in all experiments to obtain accurate estimates of predictive performance.

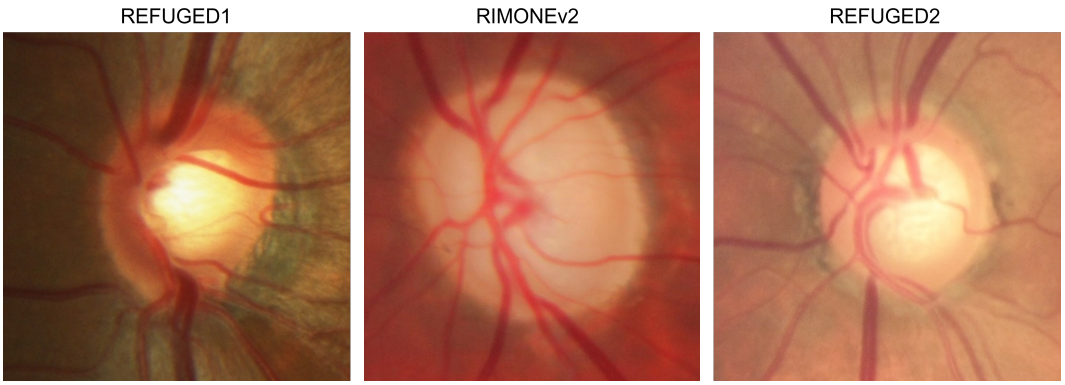


Figure 5.1: Images captured through Zeiss Visucam 500 (Left), Nidek AFC-210 fundus camera with a body of Canon EOS 5D Mark II (centre), Canon CR-2 (Right).

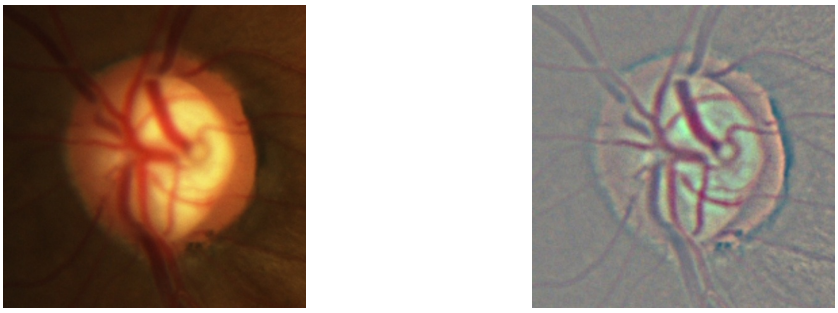
5.1 Preprocessing methods used in existing research

Xiong et al. [27] have proposed a method named enhanced domain transformation to improve the problem of domain generalisation. The idea is that if a transformed colour space presents the test data with an identical distribution as the training data, the model formed should become more generalised. We adopt this idea in our experiments by applying the matching of histograms to test data using colour histograms computed using training data.

In addition, colour normalisation is a technique that can be used to process retinal fundus images. A study conducted by Goatman et al. [79] concluded that histogram matching is more effective than grey world normalisation and histogram equalisation in the identification of four lesion types in diabetic retinopathy screening.

In their study of the automatic detection of rare pathologies in fundus photos, Quellec et al. used an image preprocessing technique to reduce dependence on the device. First, the images are converted into YC_bC_r colour space where Y is the luminance component and C_b, and C_r are the blue-difference and red-difference chroma components. Then the Gaussian kernel is used for

background removal in Y channel [2]. A blurred image is eliminated from the Y channel that is more or less equal to mean removal. Their method had similar results with transfer learning in the identification of glaucoma. We adjust the method in our study by standardising the Y channel of each image to test for camera dependency.



(a) raw image (b) preprocessed image

Figure 5.2: An example image to process an image as in [2]

The median filter is a nonlinear digital preprocessing technique, often used to remove image noise [80]. Shoukat et al. used the median filter together with the Gabor filter and adaptive histogram equalisation to design an automatic method based on EfficientNet for early detection of glaucoma using fundus images [81]. Their system performed above 90% accuracy for two independently trained models using two public datasets. However, they did not test the system across devices, which is a limitation of the study. In contrast, we use median filtering with image standardisation as a preprocessing technique by testing across each device in our study.

5.2 Methodology

Following are the three preprocessing methods we select for domain generalisation in fundus image classification to identify glaucoma.

5.2.1 Median Filtering

Median filtering is a noise removal method commonly used in image processing. More precisely, the median filter efficiently eliminates noise while preserving the edges, especially the noise of salt and pepper. The algorithm works by moving one window per pixel in the image and replacing each pixel value with the median pixel value of its neighbours. For 2D data like images, the window should include all entries in a given radius or ellipsoid region—the greater the radius, the greater the smoothness. The first step is to compute the median by sorting all pixel values in the window in decreasing or increasing order. Next, it replaces the relevant pixel value with the median (mean) value [82]. We applied this technique before applying the input standardisation, which will be described in more detail in the 5.3.2.2 section.

5.2.2 Input Standardisation

The normalisation of appearance can be achieved by alleviating variations in illumination across an image. This transformation can be carried out effectively in YC_bC_r colour space. C_b and C_r are the blue-difference and red-difference chrominance information, respectively, whereas the Y component holds the luminance. Only the Y channel can be normalised while the other two channels remain unchanged to respond to lighting variations. [2] describes a method that uses a large Gaussian kernel to estimate the background of the image and remove it from the Y channel to allow device-independent analysis. Then the image is converted into RGB colour space.

Instead of subtracting a blurred version of the illumination channel, we tweak this method by standardising the Y channel pixel values. The former method removes features such as blood vessels from the optic cup, which may result in a decrease in precision when compared to our way of standardisation. In our approach of standardisation, an RGB image is first converted to the YC_bC_r colour space and then divided into three channels.

Afterwards, the illumination (Y) is standardised using

$$Y' = K\frac{(Y - \overline{Y})}{\sigma(Y)} + C \tag{5.1}$$

where Y' is the standardised illumination; K and C are scalar constants and, $\sigma(Y)$ and $\overline{Y}$ refer to the standard deviation and the mean Y value respectively. All classifications in this research were conducted in the RGB color space, so to maintain consistency, we converted the input images to RGB after the transition. The method's applicability will be elaborated in Section 5.3.2.3.

5.2.3 Histogram Matching

Histogram matching, also known as histogram specification, is a technique for generating images with a defined histogram [80]. It modifies the red, green, and blue histograms independently to match the shapes of three specified histograms from the channels of a certain target image. The primary objective of histogram matching is to make the histogram-based statistics for an image, such as exposure and colour distributions, identical to those of a reference image, hence improving the effectiveness of some image processing algorithms.

Let f be a channel in the target image and g be a channel in the reference image. The basic technique for histogram matching is as follows: The cumulative distribution functions $cdf(f)$ and $cdf(g)$ represent the cumulative probability for each image across the range 0:255. The numbers $cdf(fx)$ and $cdf(gx)$ represent the likelihood that a pixel in images f and g is less than the grey value x. To match f, create a copy f' of f and modify each grey value x in f' so that $cdf(f'x) = cdf(gx)$. In this manner, f' has the same cdf as g. If g is lighter than f, for instance, f' will be brighter than f. If g is darker than f, then f' will also be darker. Similarly, if the reference image has a greater range of grey values than the target image, the range of grey values of the target image will also be increased, etc.

5.2.3.1 Randomised Multi-image Histogram Matching

In our study, we offer a unique technique that uses histogram matching as a component: randomised multi-image histogram matching. The procedure begins with a random selection of N images from the reference set. Then, we apply histogram matching to the target image using one of the N images. The generated image serves as the target for the subsequent iteration. The two preceding steps are repeated for each reference image so that there are N transformations. The iterative technique will transfer the overall appearance of the target image in numerous steps, resulting in a composite histogram of multiple reference images. Algorithm 5.1 depicts the proposed approach, while Figure 5.4 provides an illustration of its use.

Algorithm 5.1 Randomised multi-image histogram matching

Input : I_T, a single target image drawn from the unseen domain test set

Input : TR, the set of training images from the seen training domain

Input : N, $N \in \mathbb{Z}^+ \ll |TR|$

$H \leftarrow \{I_{TR}^1, I_{TR}^2, I_{TR}^3, ..., I_{TR}^N\} \subset TR$; `// random subset`

$I_{TT}^0 \leftarrow I_T$ **for** $i = 1 \ldots N$ **do**

  $I_{TT}^i = \text{matchHistogram}(I_{TT}^{i-1}, I_{TR}^i)$

end

return I_{TT}^N, the transformed target image

5.3 Experimental setup

We evaluate the accuracy of glaucoma classification using a single and combinations of the preprocessing techniques described in Section 3. The procedure included four steps: (1) selection of datasets, (2) selection and use of preprocessing techniques, (3) feature extraction (4) training and testing of models.

Table 5.1: Pipelines of preprocessing methods as train and/or test time

Method	Train image preprocessing	Test image preprocessing
(A) Standardisation	Standardisation	Standardisation
(B) Histogram matching	NA	Histogram matching
(C) Median filtering before standardisation	Median filtering and standardisation	Median filtering and standardisation
(D) Method (C) and histogram matching	Median filtering and standardisation	Histogram matching
(E) Standardisation and histogram matching	Standardisation	Histogram matching
Baseline	No preprocessing	No preprocessing

5.3.1 Datasets

The REFUGE and RIMONEr2 datasets were chosen for our study because their photographs are labelled and publicly accessible for research. Detailed descriptions of the two datasets are available in Sections 2.4.1 and 2.4.2. Moreover, we refer to the subsets of images collected by the two devices as REFUGED1 and REFUGED2, respectively. In our tests, we employed the ground truth labels (glaucoma/non-glaucoma) given by the REFUGE challenge Task 1 reference standard [30].

5.3.2 Image Preprocessing

First, using the method described in [76], we centre cropped photos of the REFUGE dataset around ONH to create images of size 480px by 480px. RIMONEr2 images are already center-cropped around the optic nerve head (ONH). To ensure consistency during testing, REFUGE images were similarly center-cropped. Then, we used image standardisation, histogram matching, and median filtering to the train/test data, as itemised in Table 5.1.

Figures 5.3, 5.4, and 5.5 display a set of sample images in accordance with methods (A), (B) and (C), respectively.

5.3.2.1 Parameters of preprocessing methods

5.3.2.2 Median filtering

A series of tests was undertaken with a median filter applied in the RGB colour space for each of the three channels individually, i.e. three times per image.

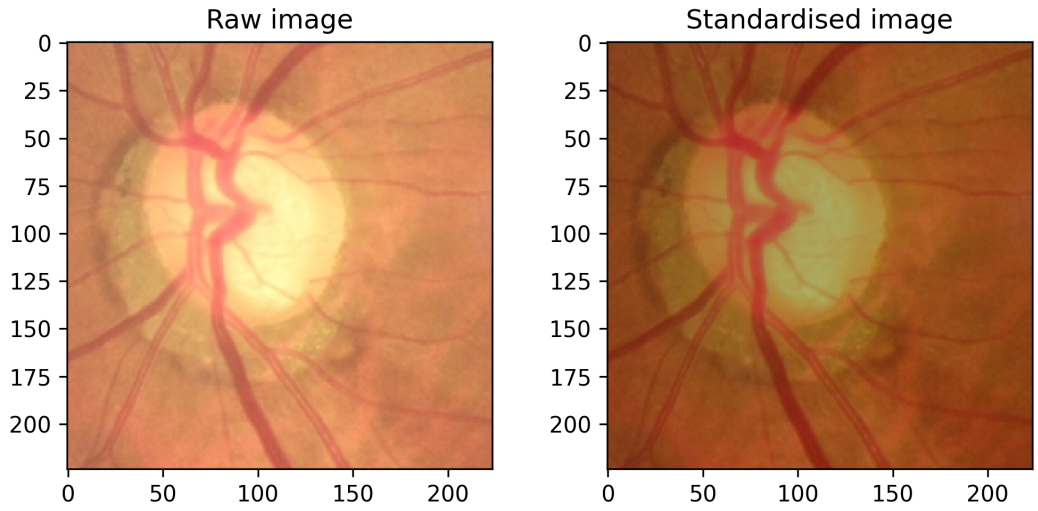


Figure 5.3: Sample image before (left) and after (right) input standardisation: Method(A).

The neighbourhood for the filter was picked from a window in the form of a flat disc with a radius of five pixels.

5.3.2.3 Input standardisation

Using the Equation 5.1, images were standardised with the proposed technique. Both K and C are variable scalar and additive constants, and we set their values to 20 and 100, respectively. Values were obtained in repeated pilot experiments using the trial and error method.

5.3.2.4 Multi-image histogram matching

Using histogram matching, we structured our experiments to normalise the texture of test images to that of the training set. First, ten reference photos per test image were selected at random from the training set. Subsequently, for each test image, the histogram was updated with the histogram of a reference image, and the procedure was done nine more times in succession. This step was conducted for each of the image's three channels. The number of reference images *(N)* was determined by experimentation with various values such as 1, 2, 3, 5, 10, 15. Greater *(N)* values resulted in memory errors. Higher *(N)* values

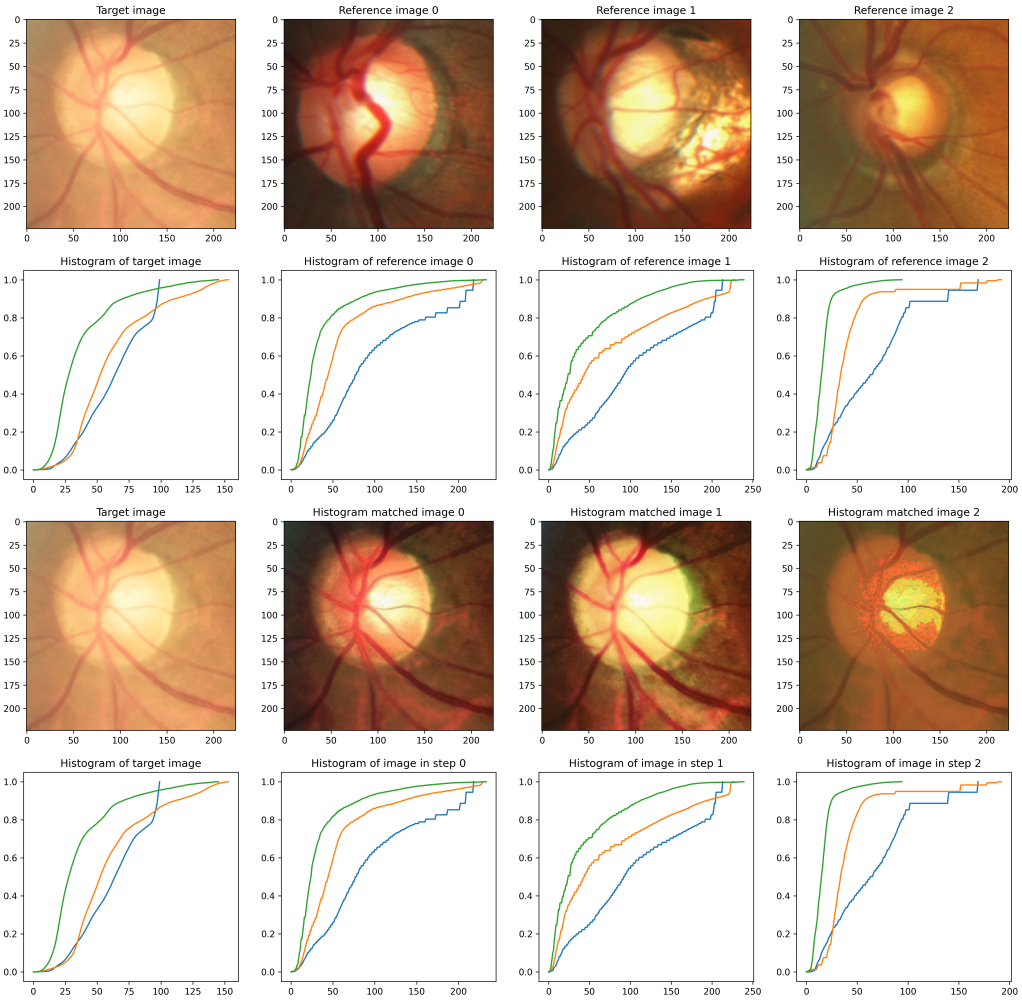


Figure 5.4: Top two rows: Target image (left, I_T in Algorithm 5.1) and three random reference/ source images with histograms of each image ($I_{TR}^1 \ldots I_{TR}^3$ in the algorithm). Bottom two rows: Target image (left) and intermediate images created as Algorithm 5.1 executes. Shown are images $I_{TT}^1 \ldots I_{TT}^3$ according to the algorithm, and if $N = 3$, then the right, lowest image is the final output transformed image.

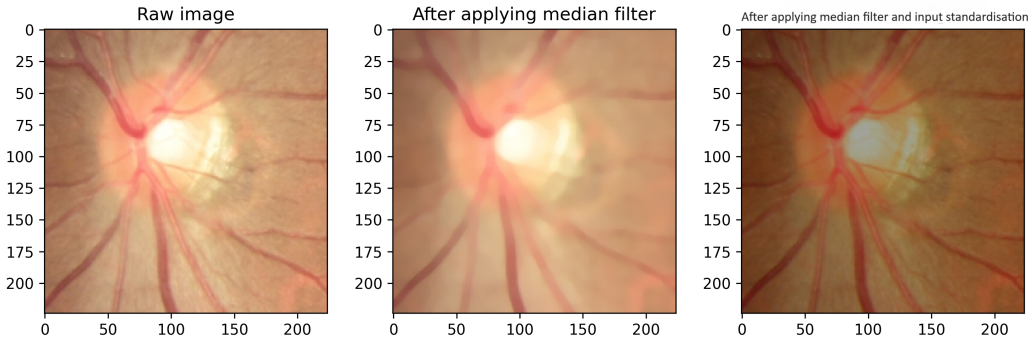


Figure 5.5: Sample image before (left) and after applying median filter (centre) before input standardisation (right): Method(C).

also increase the processing time and may distort the text image, potentially removing important features.

5.3.3 Network training and testing

After preprocessing according to one of the five techniques described in Table 5.1, features are extracted prior to learning using ResNet101V2 trained on ImageNet [75]. On the basis of the findings of the preliminary study, we use ImageNet-trained ResNet101V2 as the feature extractor. Particularly, the last 1000-class classification layer of the original network was replaced with a 2D max pooling layer. Each image is analysed to extract 2048 features by the network. Following the completion of feature extraction, we performed model training and testing using the random forest as the classifier based on the best results obtained in the preliminary investigation. As proposed by the authors of the publication [76], 1000 estimators and ten maximum features were utilised to determine hyper-parameters while building the random forest model. "Maximum features" is the number of features to consider at each split. Importantly, we modify weights inversely proportionate to class frequencies in order to balance the input data.

With each of the aforementioned image preprocessing techniques, we conducted three types of experiments: (1) train on images from a single device

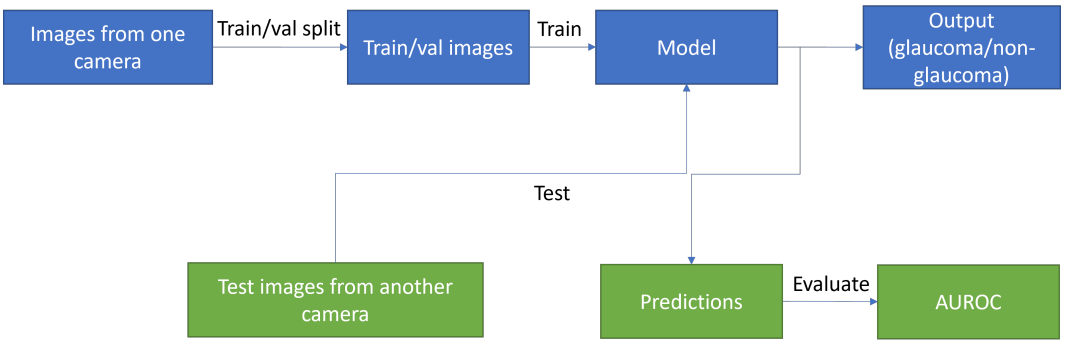


Figure 5.6: Advanced method of train/validation/test sets using two cameras.

and test on images from the other two devices separately; (2) set images from two devices as the training set and test on the third device; and (3) train and test on images from the same device as an "upper bound" for accuracy. Henceforth, they are named Experiment 1, Experiment 2 and Experiment 3, respectively. The overall data flow is shown in 5.6. Finally, we predicted the class of glaucoma for each trial. The AUROC value for the test set was obtained.

5.4 Results and Discussion

The outcomes of our experiments are summarised in Table 5.2. The first six rows reflect Experiment 1, while the middle three rows display the outcomes of Experiment 2. The lower three rows of the table display the upper limit baseline findings, which is Experiment 3. REFD1 stands for REFUGED1, and REF2 signifies REFUGED2. The dataset RIMONEr2 is indicated by the notation RIM1. In addition, the rows of Table 5.1 match with the columns of Table 5.2.

In Experiment 1, the model trained on REFUGED1 and tested on REFUGED2 had the greatest AUROC of 0.8870. When the input images are standardised, this value is reached. In addition, this technique produces the best AUROC (0.7501) for the RIMONEr2 test set trained on REFUGED1. This approach

Table 5.2: Experimental AUROCs of preprocessing methods with random forest classifier
(key: REF1=REFUGED1, REF2=REFUGED2, RIM1=RIMONE)

Exp	Train	Test	A	B	C	D	E	Baseline
(1)	REF1	REF2	**0.8870**	0.7428	**0.8870**	0.6466	0.7169	0.7949
	REF1	RIM1	**0.7501**	0.6961	0.7417	0.6916	0.6680	0.6923
	REF2	REF1	**0.8414**	0.7300	0.8030	0.6647	0.7041	0.6984
	REF2	RIM1	0.6272	**0.8488**	0.6040	0.5936	0.6227	0.5159
	RIM1	REF1	0.9252	0.7415	**0.9257**	0.7138	0.7836	0.6760
	RIM1	REF2	0.7940	**0.8585**	0.8083	0.6480	0.6699	0.5150
	Average		**0.8041**	0.7696	0.7949	0.6597	0.6849	0.6487
(2)	REF2+RIM1	REF1	0.8803	0.7730	**0.8807**	0.7206	0.7339	0.6919
	REF1+RIM1	REF2	**0.8818**	0.8059	0.8781	0.6155	0.7160	0.7071
	REF1+REF2	RIM1	0.7306	**0.8546**	0.7364	0.6214	0.7232	0.6135
	Average		0.8309	0.8112	**0.8317**	0.6525	0.7243	0.6708
(3)	REF1	REF1	0.8030	0.8247	0.7708	0.6771	**0.9722**	0.7595
	REF2	REF2	0.9405	0.9410	**0.9457**	0.7196	0.7669	0.9436
	RIM1	RIM1	0.9627	0.9216	**0.9706**	0.7828	0.8745	0.9368
	Average		**0.9021**	0.8957	0.8957	0.7265	0.8712	0.8800
Overall Average			**0.8457**	0.8255	0.8408	0.6796	0.7602	0.7332

yields AUROC 0.8041 on average, which is the greatest among all methods. This technique of standardisation delivers the greatest results when trained on a single device. However, standardisation fails when a model is trained on REFUGED2 and evaluated on RIMONEr2 with an AUROC of 0.6272: histogram matching, however, considerably improved the results.

Experiment 2 was conducted by using training images from two devices and testing images from a third device. When evaluated on REFUGED1 and trained on the other two devices, the model achieves an AUROC of 0.8807. Prior to standardising the input images in training and testing, we performed a median filtering step. The average AUROC is 0.8317, which is an improvement over Experiment 1 and the greatest average across all approaches. When there are a greater number of datasets, this unique combination yields the greatest

results.

In addition, we ran Experiment 3 to train and test using each approach on the same device. Standardisation of the inputs with a median filter prior yields corresponding AUROCs of 0.9457 for REFUGED2 and 0.9706 for RIMONEr2. The test AUROC is 0.9722 when the train data are standardised, and test data are histogram-matched on REFUGED1, suggesting that this setup is biased towards REFUGED1. Given that the dataset comprises only 400 images, of which merely 40 are categorised under glaucoma, these are distributed in an 80/20 split between the training and test sets. Consequently, multi-image histogram matching repeatedly utilises the same images as reference, thereby reducing randomness. However, the best AUROC average (0.9021) was generated for standardisation.

If images are not preprocessed, performance is severely impaired. In this situation, Experiment 1 and Experiment 2 acquire AUROC values of 0.6487 and 0.6708, respectively, which are lower than the values given by all preprocessed scenarios. Even when training and testing on the same device, the total average baseline AUROC is 0.8800, which is the second-lowest accuracy in this setting.

Standardising the input images is the optimal setup to conclude. It provides a higher average AUROC score than the results obtained by the random forest classifier for other settings. Additionally, applying median filtering before standardisation improves outcomes. The findings of Experiment 1 demonstrate that the histogram matching approach works better for RIMONEr2 and REFUGED2. The outputs of Section 5.4 were recently published with the title of "Domain generalisation for glaucoma detection in retinal images from unseen fundus cameras" [83].

5.5 Extended experiment

Boosting is a solid alternative for bagging. Instead of aggregating predictions, boosters turn weak into strong learners by concentrating on where the individual models went wrong. Here, individual models train upon the difference between the prediction and the actual results, also known as the residuals. Rather than aggregating trees, gradient-boosted trees learn from errors during each boosting round.

XGBoost shortens "eXtreme Gradient Boosting." The term "eXtreme" denotes speed enhancements such as cache awareness and parallel computing, making the XGBoost nearly ten times faster than conventional gradient boosting. XGBoost includes a unique algorithm to find the best splits that optimise trees and an in-built regularisation for reducing overfitting. In general, XGBoost is a faster and more accurate version of Gradient Boosting.

XGBoost has been outperforming random forests in many applications recently, and we are considering performing experiments to evaluate its performance in glaucoma classification against device dependency. The experiment setup is similar to that of Subsection 5.3 with two alterations where the final layer of the feature extractor is replaced with an average pooling layer, and the classifier is replaced with the XGBoost. In contrast to max pooling, which has the potential to discard valuable information by exclusively selecting the maximum value, average pooling preserves a summary of the entire patch, thereby potentially retaining pertinent contextual details that might otherwise be overlooked. Table 5.3 shows the outcome of the experiments.

In experiment (1), the highest AUROC of 0.8890 is given when the model is trained on REFUGED1 and tested on REFUGED2. This value is when the input images are applied with median filtering before being standardised. Furthermore, this method gives the best AUROC (0.8729) for the RIMONEr2 as the training set tested on REFUGED1. The standardisation method gives AUROC 0.8144 on average, which is the highest among others. This standardisation method gives the best results when trained on a single device. However,

Table 5.3: Experimental results of preprocessing methods with XGBoost classifier

Exp	Train	Test	A	B	C	D	E	Baseline
(1)	REF1	REF2	0.8560	0.6946	**0.8890**	0.6906	0.6121	0.8182
	REF1	RIM1	**0.7818**	0.6303	0.7481	0.6954	0.5893	0.7063
	REF2	REF1	**0.8790**	0.7361	0.7667	0.6865	0.7378	0.7321
	REF2	RIM1	0.6614	**0.8333**	0.6153	0.5994	0.5513	0.4252
	RIM1	REF1	0.8418	0.7313	**0.8729**	0.7762	0.7370	0.8503
	RIM1	REF2	**0.8663**	0.8128	0.8135	0.7414	0.5773	0.6486
	Average		**0.8144**	0.7397	0.7842	0.6982	0.6341	0.6968
(2)	REF2+RIM1	REF1	0.8458	0.7483	0.8376	0.7834	0.7667	**0.8840**
	REF1+RIM1	REF2	**0.8788**	0.8060	0.8402	0.6829	0.6101	0.7665
	REF1+REF2	RIM1	0.7551	**0.8123**	0.7360	0.6119	0.6625	0.6347
	Average		**0.8266**	0.7889	0.8046	0.6927	0.6798	0.7617
(3)	REF1	REF1	0.9948	0.8628	**1.0000**	0.7740	0.7517	0.9670
	REF2	REF2	**0.9870**	0.9427	0.9852	0.8056	0.6970	0.9701
	RIM1	RIM1	**0.9578**	0.8686	0.9534	0.6451	0.7936	0.9554
	Average		**0.9799**	0.8914	0.9796	0.7416	0.7475	0.9642
Overall Average			**0.8736**	0.8067	0.8561	0.7108	0.6871	0.8075

standardisation fails when a model is trained on REFUGED2 and tested on RIMONEr2 with an AUROC of 0.6614.

Experiment (2) was performed by setting up training images from two devices and testing on the third device. The model gives testing 0.8840 AUROC when tested on REFUGED1 and trained on the other two devices but without any preprocessing method applied. The average AUROC is 0.8266, an improvement from the experiment (1), which is also the highest average given by standardisation compared to other methods.

Furthermore, we conducted the baseline experiment (3) to train and test on the same device using every method. Standardising the inputs gives 0.9870 and 0.9578 AUROC for REFUGED2 and RIMONEr2, respectively. Standardising the train data with the median filter prior on REFUGED1 sets the test

AUROC for 1.0000. However, the best average AUROC (0.9799) was given for the standardising.

In summary, standardising the input images is the best configuration. It gives an overall higher average AUROC score compared with the scores returned for other settings by the XGBoost classifier. Also, applying median filtering prior to standardising gives better results. Furthermore, the Histogram matching method is biased towards RIMONE and REFUGED2, as shown by the results in experiments (1) and (2).

Comparing the classification results of random forest and XGBoost, the input standardisation preprocessing method outperforms both classifiers. Importantly, the overall best average was increased in XGBoost compared to random forest. Also, median filtering applied prior to standardisation achieves comparable results in both classifications. However, preprocessing methods that involve histogram matching failed with XGBoost classification, indicating that the choice of preprocessing method and the classifier may have a correlation.

The preprocessing of an image, which involved a combination of multi-image histogram matching, input standardization, and median filtering, required approximately five seconds. The durations for feature extraction and classification were comparable to those reported in Chapter 4.

Chapter 6

Surrogate optimisation of deep stacked transformation

This chapter uses deep stacked transformation (DST) as an augmentation method. Existing literature limits the use of single transformation or a few random transformations when augmenting images in glaucoma classification. However, by utilising DST, we introduce a proper method for selecting a sequence of transformation operations and their parameters through optimisation techniques.

6.1 Methodology

6.1.1 Deep stacked transformation

Deep stacked transformation, abbreviated DST, was originally introduced by Zhang et al. to segment MRI and ultrasound images [84]. They gathered several datasets from one domain and trained and validated their approach using only one dataset. Finally, they tested against each of the remaining datasets. Prior SOTA results were included for comparison, and the baseline results with no DST applied were also included. Their results have achieved SOTA, which implies that their method is successful in domain generalisation in medical image segmentation.

DST is a sequence of n stacked transformation as shown in Algorithm 6.1. Each transformation is an image processing function with two parameters: (1) the probability p of applying the function and (2) the magnitude m of the function.

Algorithm 6.1 Deep stacked transformation

Input: x

Output: $x_n = transform(x)$

Function `transform(x)`:

$x_1 = operation_1(x, p_1)$

$x_2 = operation_2(x_1, p_2)$

.

.

.

$x_n = operation_n(x_{n-1}, p_n)$

return x_n

End Function

The original DST research paper used nine transforms: three from each of image quality (sharpness, blurriness, and noise level), appearance (brightness, contrast, and intensity perturbation), and spatial configuration (rotation, scaling, and deformation). The authors of the above study performed all the operations with an equal probability of 0.5, but they state that the probabilities can be the same or different for each operation. Magnitudes and/or their ranges were chosen based on the transformation. Also, the order of transforms does not have to be fixed.

We hypothesise that applying DST will increase the cross-domain glaucoma classification accuracy. Hence, we apply a set of transformations in a sequence, first with default 0.5 probability and then experiment on optimising the probability of each operation using different approaches such as local search and surrogate optimisation.

6.1.2 Local search optimisation

Local search is a heuristic method for solving computationally hard optimisation problems. It can be used on problems that are formulated for finding a solution maximising a criterion from a number of candidate solutions. Local search algorithms explore the search space by applying local changes until an optimal solution is found or exceeds a given time constraint.

Hill-climbing is a mathematical optimisation technique that is a type of local search. The iterative algorithm starts with a random solution (or predefined default solution) to a problem and tries to find a better solution by making an incremental change to the current solution. If it succeeds, the new solution is incrementally changed, and so on, until no further improvements can be found or if it exceeds the maximum number of iterations.

In our experiments, we used the hill-climbing approach to find the optimal probabilities of the DST operations. First, we tried systematically incrementing the probability by 0.1, starting from 0 to 1. However, lower values resulted in poor performance of glaucoma classification. Hence, we start at all the probabilities set to 0.5 and update it by adding Gaussian noise. The pseudo code for the algorithm is presented in Algorithm 6.2.

After trying different numbers such as 1, 20, 50, 100 and 150, the number of iterations was set to 100, where 150 caused a memory error. The datasets used in this study varied in the number of images they contained. Consequently, the number of iterations was carefully selected to accommodate all datasets. Furthermore, the experiments were conducted using shared remote servers and online platforms, such as Google Colaboratory[1], to facilitate parallel processing. This setup necessitated choosing the minimum number of iterations necessary to achieve the highest possible accuracy, considering the constraints on network bandwidth and memory availability.

[1] `https://colab.research.google.com/`

Algorithm 6.2 Policy optimisation using local search algorithm

Input: data $->$ trainData from device1, valData from device2

Input: policy $->$ list of operations with magnitudes and probability=0.5

Output: best $<-$ policy5

local variables: model $<-$ model to be trained, initially trained on ImageNet

Function `search(`*data, policy*`):`

```
VALAUC[best] <- train and evaluate on best
for repeat 100 times do
    current <- best with added random noise to probability using Gaussian distribution
    train the model with current policy
    trained on augmented train data
    validated on unaugmented val data
    pick the best model with the highest val_auc
    VALAUC[current]
    if VALAUC[current] > VALAUC[best] then
        best <- current
end
return best
```

End Function

6.1.3 Surrogate optimisation

A surrogate is a function that approximates another function. It attempts to find a global minimum of an objective function within a few evaluations of the function. The algorithm tries to balance the optimisation process between exploration and speed. Former searches for a global minimum, while the latter obtain a good solution in a few objective function evaluations. The stopping criterion that stops the solver near a global solution can be set to a number of function evaluations, a maximum allowed CPU time, or a maximum number of failed iterative improvement trials and take the best solution found within the computational limit.

Surrogate optimisation algorithms, in general, consist of four components:

(1) optimisation problem, (2) experimental design, (3) surrogate model and (4) strategy. The optimisation problem consists of all the available information about the problem, such as dimensionality, variable types, objective function, etc. Experimental design, on the other hand, generates the initial set of points for building the initial surrogate model. A surrogate model is used when an outcome of interest cannot be easily computed, so an approximate mathematical model of the outcome is used instead. It approximates the underlying objective function using models such as radial basis function (RBF), SVM, artificial neural networks (ANN), Bayesian networks, etc. Finally, the strategy refers to the algorithm that chooses new evaluations after the experimental design has been evaluated [85]. The general flow of a surrogate optimisation algorithm can be given as follows[2].

Inputs: Optimisation problem, Experimental design, Optimisation strategy, Surrogate model, Stopping criterion

1. Generate an initial experimental design
2. Evaluate the points in the experimental design
3. Build a Surrogate model from the data
4. Repeat until the stopping criterion met
   (a) Use the strategy to generate new point(s) to evaluate
   (b) Evaluate the point(s) generated using all computational resources
   (c) Update the Surrogate model

Outputs: Best solution and its corresponding function value

6.2 Experimental setup

We designed our experiments to test DST's generalisability while optimising the transformations' probabilities. We selected 12 transformations and ranges

[2] `https://pysot.readthedocs.io/en/latest/surrogate_optimization.html`

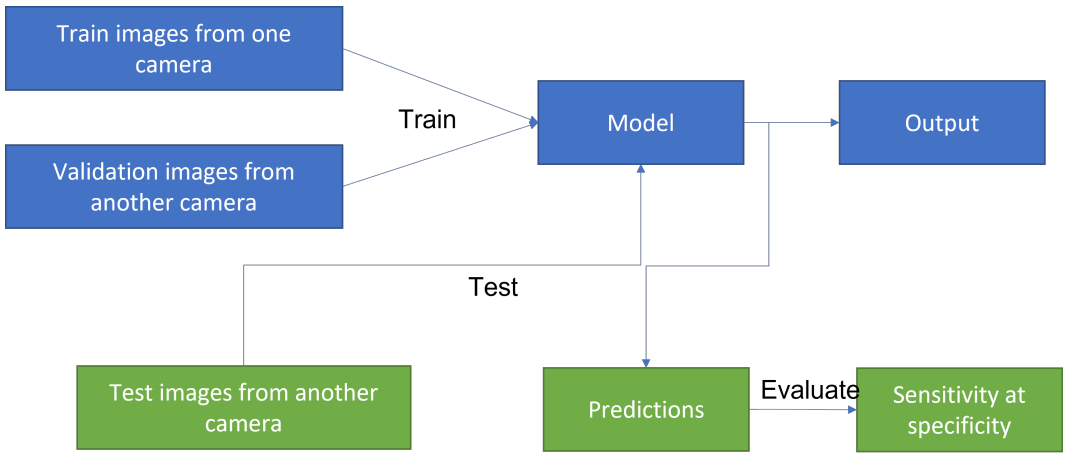


Figure 6.1: Proposed method of train/validation/test sets using three cameras.

of magnitudes by studying relevant research, expert suggestions and pilot testing[3].

We used the same datasets and the initial cropping around ONH as described in Section 5.3.1. The described datasets include images from three different fundus cameras. Hence, we considered images from a certain device to be the training set, another one to be the validation set and the remaining one to be the test set, as shown in Figure 6.1. We repeated the training process for each combination of three cameras, six in total. Furthermore, we monitored the sensitivity at the specificity of the validation set to report the performance.

6.2.1 Tools

We used several machine learning and related software packages when performing the experiments based on the compatibility and easiness of using them. TensorFlow 2.10 [86] was used as the machine learning library to train the models. Albumentations library [87] and pySOT [85] were used as DST and surrogate optimisation libraries.

[3] `https://demo.albumentations.ai/`

6.2.1.1 Albumentations

Data augmentation is a set of techniques that increase the size and quality of training datasets so that better deep learning models can be trained using them [88]. Albumentations is a widely used library to augment computer vision and deep learning-related research areas. It is a fast image augmentation library and easy to use wrapper around other libraries. Fast augmentations can be done as it is based on a highly-optimised OpenCV library. Simple but powerful interfaces and modules are for separate tasks like classification, detection, segmentation, etc. Furthermore, it is easily customisable and easy to add other frameworks such as TensorFlow and PyTorch to the library.

There are more than 100 transforms that belong to one of the pixel-level or spatial-level transforms. Some are included in the domain adaptation module, some are in the functional module, and the rest are in the transforms module. We used 12 transforms altogether: Contrast Limited Adaptive Histogram Equalization (CLAHE), rotation, transpose, fancy PCA, randomly changing the hue, saturation and value of the input image, horizontal flip, channel dropout, emboss, random change of brightness and contrast, inverting the image, RGB shift where randomly shifting values for each channel of the input RGB image, and image normalisation.

The list of transforms and magnitudes are summarised in Table 6.1. We arbitrarily selected the number of transforms which exceeds the one in [84]. Despite lacking parameter values, the transformations Transpose, HorizontalFlip, and InvertImg were selected due to their practical implications. The Transpose transformation can alter the orientation of an image from portrait to landscape, or vice versa. This is particularly beneficial for ensuring that images adhere to the expected input size and orientation of a neural network architecture. HorizontalFlip is generally less disruptive; it mirrors the image content horizontally without altering its vertical structure. InvertImg, on the other hand, creates high-contrast visuals. This can be advantageous in applications such as devices that offer a dark mode, where image inversion may

Table 6.1: DST transforms and parameter values as defined in Albumentations library. The value of the parameter "always_apply" of every transform was set to False with the value of "p" was changed during the execution of the DST.

Transform	Parameter values
CLAHE	clip_limit=4.0, tile_grid_size=(8, 8)
Rotate	limit=20
Transpose	Transpose the input by swapping rows and columns
FancyPCA	alpha=0.1
HueSaturationValue	hue_shift_limit=20, sat_shift_limit=30, val_shift_limit=20
HorizontalFlip	Flip the input horizontally around the y-axis
ChannelDropout	channel_drop_range=(1, 1), fill_value=0
Emboss	alpha=(0.2, 0.5), strength=(0.2, 0.7)
RandomBrightnessContrast	brightness_limit=0.2, contrast_limit=0.2, brightness_by_max=True
InvertImg	Invert the input image by subtracting pixel values from 255
RGBShift	r_shift_limit=20, g_shift_limit=20, b_shift_limit=20
Normalize	mean=(0.485, 0.456, 0.406), std=(0.229, 0.224, 0.225), max_pixel_value=255.0

help in reducing glare.

6.2.1.2 pySOT

Surrogate Optimization Toolbox (pySOT) is designed for global deterministic optimisation problems. The toolbox's main purpose is to optimise computationally expensive black-box objective functions with continuous and/or discrete variables. All variables should have bound constraints in some form but could not be infinity. Tight bounds make the algorithm efficient as it reduces the search space and increases the quality of the built surrogate.

In our experiment, we use pySOT to find the optimal probability values of transforms in DST. RBFInterpolant[4] was used as the surrogate model with SRBFStrategy[5] to explore the search space between 0.0 and 1.0 (probability). The experimental design SymmetricLatinHypercube[6] generates 26 sampling points initially, calculated using 2 x 12 + 1, where 12 is the number of trans-

[4] `https://rbf.readthedocs.io/en/latest/interpolate.html`

[5] `https://pysot.readthedocs.io/en/latest/options.html#srbfstrategy`

[6] `https://pysot.readthedocs.io/en/latest/options.html#symmetriclatinhypercube`

Table 6.2: Classification accuracy after surrogate optimisation vs. local search optimisation of DST parameters

Dataset			Surrogate optimisation		Local search optimisation	
Train	Validation	Test	Validation AUROC	Test AUROC	Validation AUROC	Test AUROC
RIMONE	REFUGED1	REFUGED2	0.9225	0.9050	0.9325	0.9063
RIMONE	REFUGED2	REFUGED1	0.9337	0.9101	0.9525	0.8375
REFUGED1	RIMONE	REFUGED2	0.9440	0.9112	0.8813	0.8525
REFUGED1	REFUGED2	RIMONE	0.9287	0.8473	0.9425	0.9253
REFUGED2	RIMONE	REFUGED1	0.8945	0.8900	0.8512	0.8300
REFUGED2	REFUGED1	RIMONE	0.9400	0.9220	0.9400	0.8626

forms. The maximum number of function evaluations was set to 100 to match with local search optimisation.

6.3 Results and discussion

We reported AUROC for validation and test sets using the proposed new train/val/test set preparation method by comparing surrogate optimisation and local search optimisation. The results are shown in Table 6.2.

Comparing local search optimisation and surrogate optimisation of DST parameters, surrogate optimisation shows better classification accuracy on average, with 0.9272 and 0.8976 AUROCs for validation and test data, respectively. In contrast, the values were 0.9167 and 0.8690 for local search optimisation. These values represent an average of ten executions of the experiment.

When comparing the accuracy based on the training dataset, the classifier achieved an average test AUROC of 0.8976 in the surrogate and 0.8690 in local search optimisation methods. The highest test average AUROC of 0.9075 was shown by the model for the surrogate method with RIMONE as the training set, and the value was 0.8719 for the local search method. The model gave average test AUROCs of 0.9060 and 0.8463 when it was trained by REFUGED2 data, keeping the surrogate method the best optimisation technique. However, The model performed slightly well under local search optimisation when the training set was REFUGED1 with 0.8793 and 0.8889 as average test AUROCs

on the REFUGED2 and RIMONE datasets, but the numbers are almost the same.

The model gave a test accuracy of 0.9000 when the DST parameters were optimised by the surrogate method, but the value was reduced to 0.8338 when the local search method was used with REFUGED1 as the test set. Similarly, 0.9081 and 0.87938 were the reported values for REFUGED2 as the test set. The surrogate optimisation technique has produced better-optimised parameters than the local search for REFUGE data. However, the results swapped when the model was tested using RIMONE with AUROC values of 0.8846 and 0.8940 by respective optimisation techniques.

It should be noted that RIMONE is a nearly balanced dataset compared to REFUGE and that REFUGED2 consists of twice the data size as REFUGED1. Also, we have used two different datasets in the training phase for training and validation, respectively. Hence, the model was exposed to an unseen domain while training, creating a more complex training environment. Also, the validation dataset was not preprocessed or balanced during the training time. However, one can balance the validation dataset to see the accuracy difference. Nevertheless, it may negatively affect the unseen domain problem, where an actual clinical environment may have heavily imbalanced datasets.

In the context of memory resource utilisation and runtime performance, the experiments detailed in this chapter were conducted within an Ubuntu server environment, leveraging GPU capabilities. Specifically, the NVIDIA GeForce RTX 3080 GPU [89] was employed for the experimental procedures. Across the various experiments, the average processing time to handle one dataset and attain results stood at approximately 3.0 hours. This duration encapsulates the entirety of each experiment's execution cycle, from data preprocessing through model training to result evaluation.

Chapter 7

Neural style transfer for domain generalisation

7.1 Introduction

Neural style transfer, abbreviated NST, is an optimisation technique that blends two images; a content/target/input image p with a style reference image a. The output image x will look like the content/target image that is "painted" in the style of the reference image [90]. The algorithm can optimise the output image in two types of statistics, namely, content statistics and style statistics which come from the content image and the style reference image, respectively [90]. NST is based on texture synthesis algorithms based on histograms. They use the CNN features to perform the synthesis, where the underlying CNN is specifically trained for object detection.

The algorithm requires an input image p and an example style image a. First, the image p is provided through the CNN, and network activations are sampled at a late convolution layer of the network architecture. Assume $C(p)$ be the resultant output sample, named the 'content' of the input p. Then, the style image a is provided through the CNN, and network activations are sampled at the network architecture's early to middle convolution layers. Assume $S(a)$ be the encoded Gram matrix representation of these activations,

named the 'style' of a. Neural style transfer synthesises the output image x that shows the content of p applied with the style of a, i.e.

$$C(x) = C(p)$$

and

$$S(x) = S(a)$$

x is gradually updated iteratively by an optimisation to minimise the loss function error:

$$\mathcal{L}(x) = |C(x) - C(p)| + \alpha |S(x) - S(a)|$$

where, $|.|$ is the L2 distance. The constant α controls the level of stylisation effect [91].

Figure 7.1 shows a graphical illustration of NST algorithm. Key concepts of NST can be given as follows [90].

1. **Content Image:** The image whose overall structure and content will be preserved in the final output.

2. **Style Image:** The image whose textural elements are to be mimicked in the final output.

3. **Convolutional Neural Network (CNN):** A pre-trained network like VGG is used. The CNN is not trained to create new images, but rather to differentiate and characterise style and content from existing images.

4. **Feature Extraction:** The CNN processes both the content and style images, extracting information about the content from certain layers and style information from various layers across the network.

5. **Loss Functions:** There are two main loss functions used in style transfer:

   (a) **Content Loss:** Measures how much the content of the output image differs from the content of the content image. This is usually

calculated using the feature maps from one or more layers deep in the network where high-level features are captured.

(b) **Style Loss:** Measures the difference in style between the output image and the style reference. This involves comparing the correlations between different sets of features across the layers of the CNN, which capture the textures and colors that represent the style.

6. **Optimization:** Starting from a noise image (or sometimes the content image itself), the algorithm iteratively updates the pixels of the initial image to minimize both content and style loss. This optimizes the image to resemble the content of the content image while adopting the style of the style image.

7. **Total Variation Loss:** Sometimes an additional loss function is used to encourage spatial smoothness in the final output, ensuring that the generated image does not have too much high-frequency content which can result in excessive noise.

The NST technique has been utilised in related research for generating synthetic images or augmenting datasets in segmentation tasks [92, 93]. We propose using it as a preprocessing method. Application of NST on a sample fundus image is shown in Figure 7.2.

7.2 Methodology

7.2.1 GlaucomaNet: A Custom CNN

CNN was first introduced by LeCun et al. in 1990 [94]. The convolution block is CNN's main building block that includes several layers in one block. As shown in the CNN architecture diagram in Figure 7.3, input, convolution, and pooling layers do the feature extraction while fully connected and output layers perform the classification.

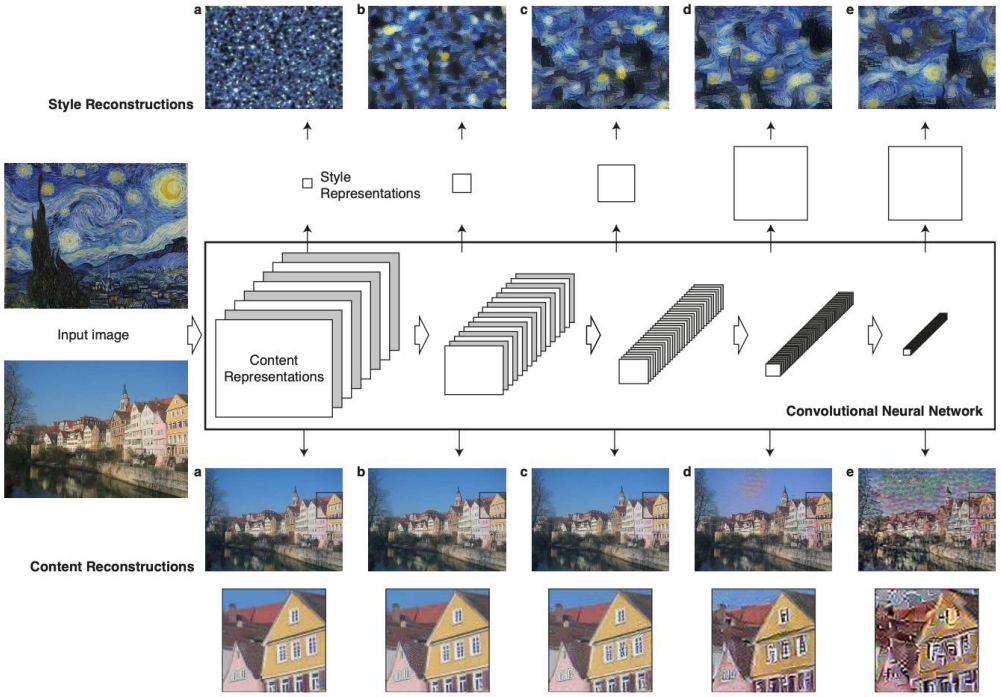


Figure 7.1: Outline of NST

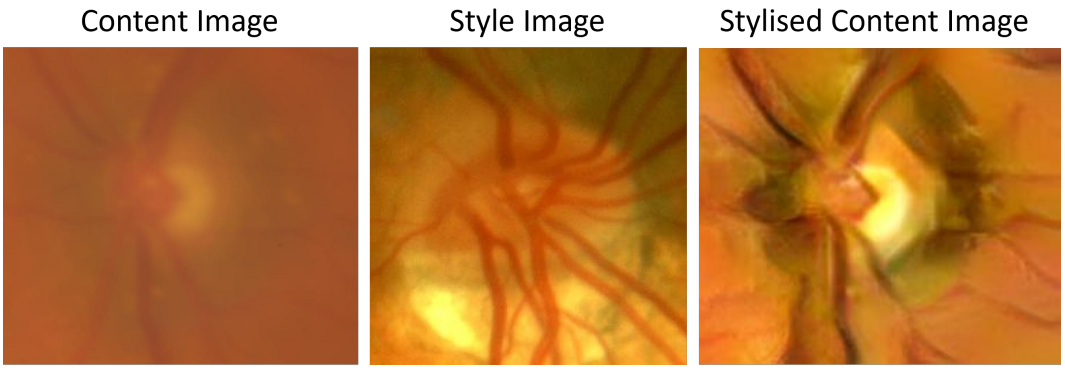


Figure 7.2: Outline of NST

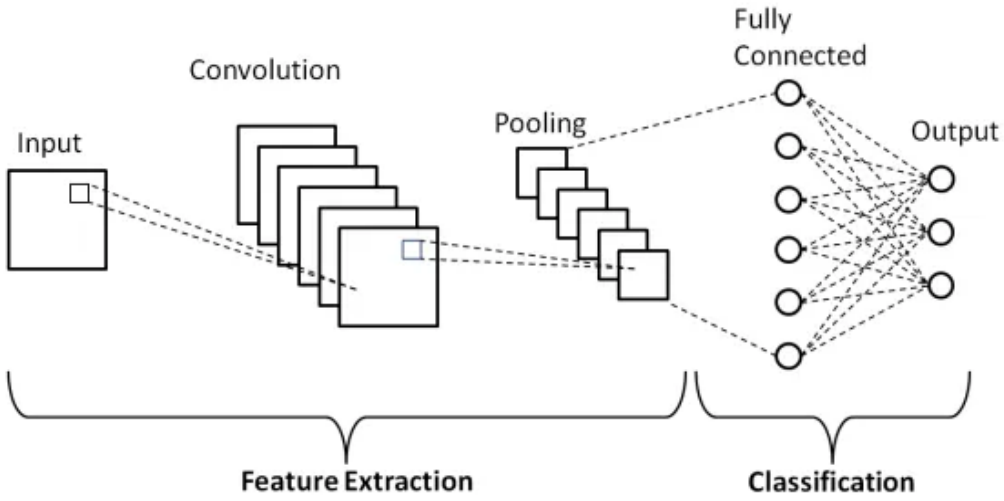


Figure 7.3: Outline of CNN

We designed a custom CNN named GlaucomaNet that performed well for the selected datasets. The first half of the network comprises the input layer followed by six convolution blocks. Each of the last five convolution blocks has a 2D convolutional layer, a batch normalisation layer and a 2D max pooling layer. There is a global max pooling layer following the convolution blocks. A fully connected layer follows it. All the convolutional and dense layers use rectified linear unit as the activation function. The output layer with the softmax function classifies the images into glaucoma and non-glaucoma classes.

Figure 7.4 shows a graphical visualisation of the model architecture. In comparison to the custom CNNs discussed in Subsection 3.1.2, the proposed model integrates a batch normalization layer within each convolutional block. Moreover, this model is capable of processing larger input sizes than those handled by related models.

7.2.2 Experimental setup

This section aims to study the effect of combining selected preprocessing methods on the unseen domain problem in glaucoma classification. The methods under consideration are (1) image normalisation, (2) deep stacked transformation and (3) neural style transfer. Furthermore, different classification meth-

ods were used, such as (1) GlaucomaNet, (2) random forests and (3) XGBoost. Decision trees are used with feature extraction by ImageNet-trained AlexNet prior to classification. Reference style images were randomly selected from the ACRIMA dataset. Moreover, the hyperparameter values of the classifiers were chosen based on previous Chapters 4 and 5.

We conducted 48 experiments with different parameters of the above methods, as shown in Table 7.1. A set of experiments were executed with and without applying image normalisation and deep stacked transformation. Each of them had NST applied with four values of alpha: the parameter which determines the transfer level. The alpha was systematically chosen as none, 0.0, 0.5 and 1.0. A higher α value increases the emphasis on retaining content details in the output image [90].

Table 7.1: Parameter values of each preprocessing method

Preprocessing method	Parameter	Values
Image normalisation	Applied	Yes/No
Deep stacked transformations	Applied	Yes(probability = 0.5)/No
Neural style transfer	Alpha	None/0.0/0.5/1.0

Together with three classifiers, there were 48 settings (2x2x4x3), and each experiment was repeated for each train/test combination of three datasets. In addition, we conducted a set of experiments with a combination of all three datasets to use them as a baseline.

We monitored the classification accuracy using AUROC. We conducted a statistical test to compare method combinations to identify interesting patterns and insights to find a proper algorithm for glaucoma classification with removed device dependency.

7.3 Results and discussion

Among many statistical tests such as Z-test, t-test, ANOVA test, and chi-square test, we selected the student's t-test to compare each classification model under the above settings as it is the most common test for comparing two methods. Also, box plots were useful for graphically visualising the AUROCs compared to different models.

7.3.1 Box plots with respect to preprocessing setup

Each figure of multiple box plots compared the test AUROC values of three classifiers for each preprocessing method combination. An individual box plot contains AUROC values for six dataset combinations under each setting. We report only the best combinations in identifying the best methods and classifiers according to the maximum AUROCs.

The results are shown in Figures 7.5 and 7.6. In plots, the orange horizontal line in each box shows the median value, and the green triangle shows the mean value. The white circle is an outlier. The x-axis represents classifiers GlaucomaNet, random forests, and XGBoost stated as cnn, rfc and xgb, respectively. On the other hand, the y-axis is the AUROC.

As shown in Figure 7.5, the GlaucomaNet gave its best classification accuracy over all six combinations of train and test data when (1) the data are not normalised, (2) deep stacked transformation applied with a probability of 0.5 and (3) neural style transfer applied at a level of 0.5. The median and mean values were above 0.8, and all the test AUROC values were above 0.7. However, the other two classifiers failed to perform at the given settings.

XGBoost, on the other hand, gave its best classification accuracy over all six combinations of train and test data when all the above preprocessing steps were not applied. Like GlaucomaNet, the median and mean values were above 0.8, and all the test AUROC values were above 0.7. However, the other two classifiers failed to perform adequately at the given settings, as illustrated in

Figure 7.6.

When compared with the other two classifiers, random forests were not performing at an acceptable level under any setting. For example, the box plots corresponding to random forests are the minimum values in both graphs above.

7.3.2 Student's t-Test

Student's t-test is another statistical test used for comparing different techniques. It tests for the null hypothesis to determine the better technique among two. We compared each classifier against every other under each setting using the t-test. The degree of freedom was chosen as five because each setting had six AUROC values in six data combinations. The dataset combinations used for training and testing are as follows: RIM1/REF1, RIM1/REF2, REF1/REF2, REF2/REF1, REF1/RIM1, and REF2/RIM1. These combinations represent different configurations in which one dataset serves as the training set and the other as the test set. The results (>1.0) are shown in Table 7.2.

Table 7.2: t-values and p-values of student's t-test

isNorm	isDST	Alpha	Classifier 01	Classifier 02	t-value	p-value
TRUE	FALSE	1	GlaucomaNet	XGBoost	1.305	0.2022
FALSE	TRUE	1	GlaucomaNet	XGBoost	1.181	0.2472
TRUE	FALSE	1	GlaucomaNet	Random Forest	1.155	0.2575
TRUE	TRUE	1	GlaucomaNet	XGBoost	1.118	0.2727
FALSE	TRUE	1	GlaucomaNet	Random Forest	1.017	0.3176

The best t-values were given when GlaucomaNet was compared with other methods. Specifically, the results of the t-test confirm the best results given by box plots under the same setup. According to the table of t-values by degrees of freedom (dof) and the desired probability that the Null should be rejected for 2-tailed t-tests in Appendix B, to verify that the null should be rejected

with a 95% probability, it requires a t-value larger than 2.045. However, the maximum t-value of 1.305 resulted in the experiment being below the threshold value. The corresponding p-value is 0.202, and the result is insignificant as the p-value is greater than 0.05.

However, the results show several limitations that reduce the classification accuracy. One example is that deep stacked transformation was applied under a default setting of 0.5 probability to keep the probability constant in various preprocessing setups. Transformations are applied in various magnitude ranges, which may be suitable for one dataset but not others. Also, the probability of 0.5 is applied for all the twelve selected transformations. Hence, the number of transformations, the selection of transformations, and their magnitudes and probabilities will impact the classification accuracy, which can be achieved by policy optimisation.

According to the results of the student's t-test, GlaucomaNet seemed to be performing well compared to XGBoost, but the test does not disprove the null hypothesis. This may be because of the less number of trials performed due to limited resources. We only performed 30 times which is better if repeated 50 or 100 times.

It is noteworthy that XGBoost did not consistently outperform Random Forest in all cases, as indicated in Table 7.2. Factors such as data noise and outliers may influence the performance of XGBoost relative to Random Forests. For instance, due to its iterative nature aimed at correcting errors from previous trees, XGBoost may be susceptible to overfitting when confronted with noisy data. Additionally, the sequential tree-building process of XGBoost may present limitations in environments with stringent computational or time constraints [95].

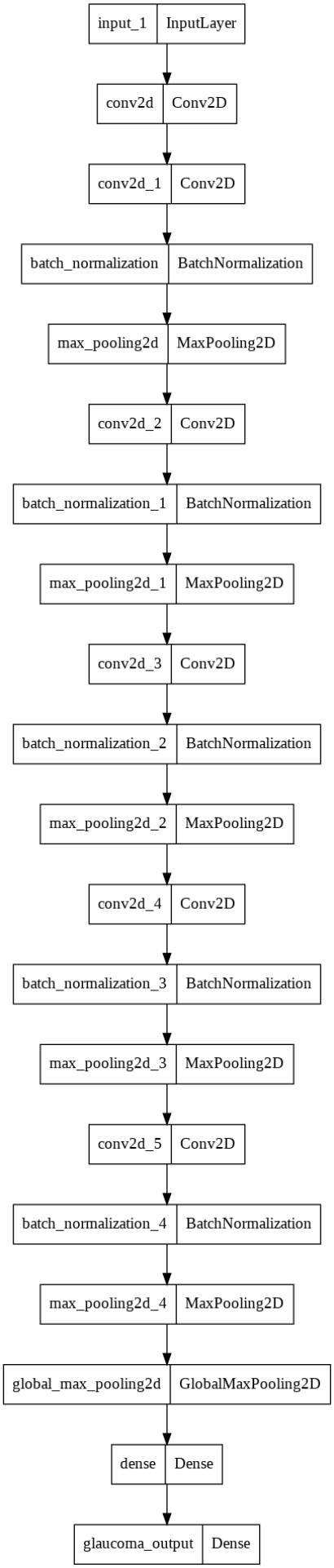


Figure 7.4: Model architecture of the GlaucomaNet

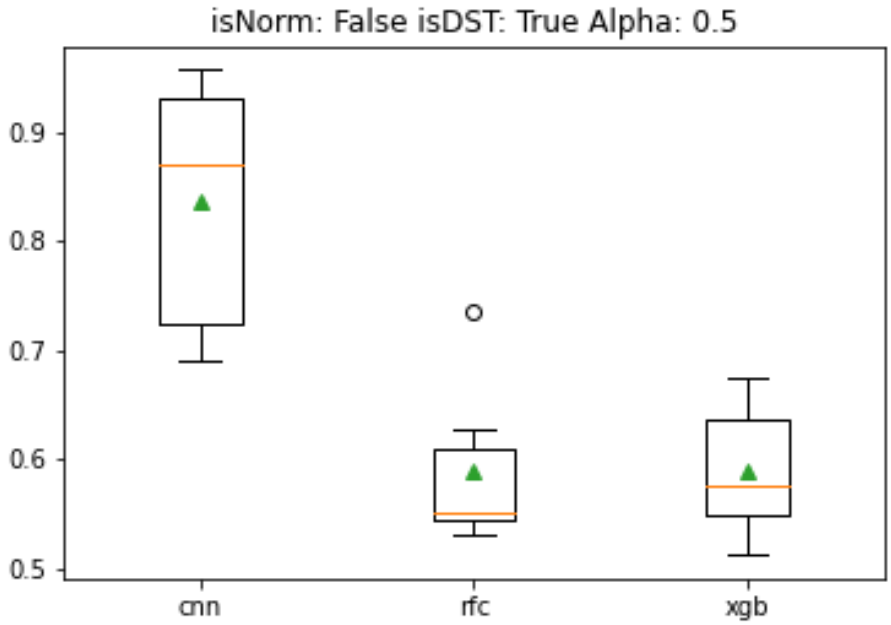


Figure 7.5: Box plot: The best input configuration for GlaucomaNet

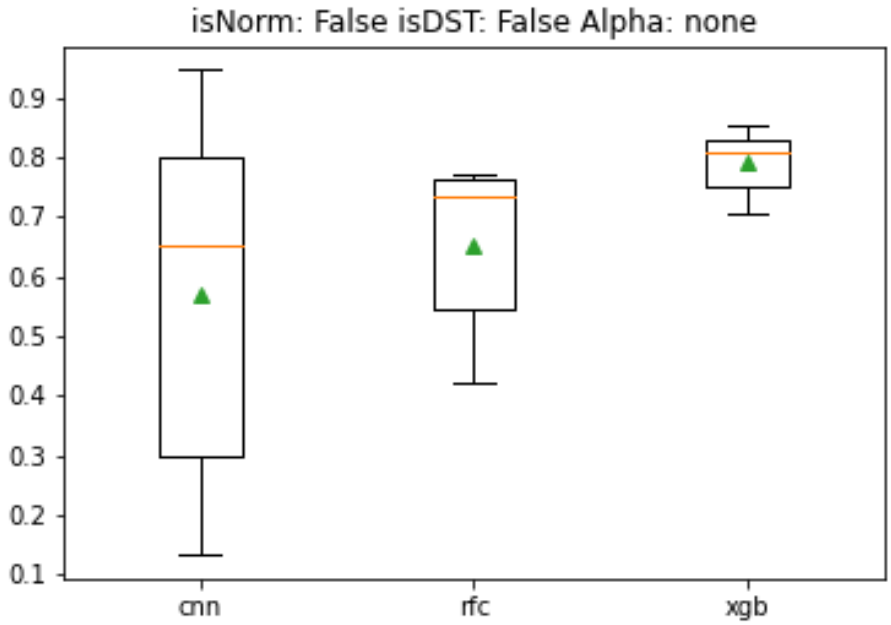


Figure 7.6: Box plot: The best input configuration for XGBoost classifier

Chapter 8

Neural Style Transferring for improving the robustness of pre-trained glaucoma classification models

8.1 Introduction

In the previous chapter, we identified NST as a potential candidate for domain generalisation in machine learning-based glaucoma classification. However, we used model training during the previous experiments. In this chapter, we aim to examine the performance of NST during the test-time augmentation of pre-trained models.

An example of applying NST to an image is shown in Figure 8.1. The first row of the figure is the raw test image selected from the REFUGE test set. The second row shows six reference images randomly picked from the ACRIMA dataset, and the first three are non-glaucoma images, whereas the last three are glaucoma ones. The last row includes all the corresponding resulting styled content images using each reference image shown in the figure's middle row.

First, we used NST as a test-time pre-processing technique to compare

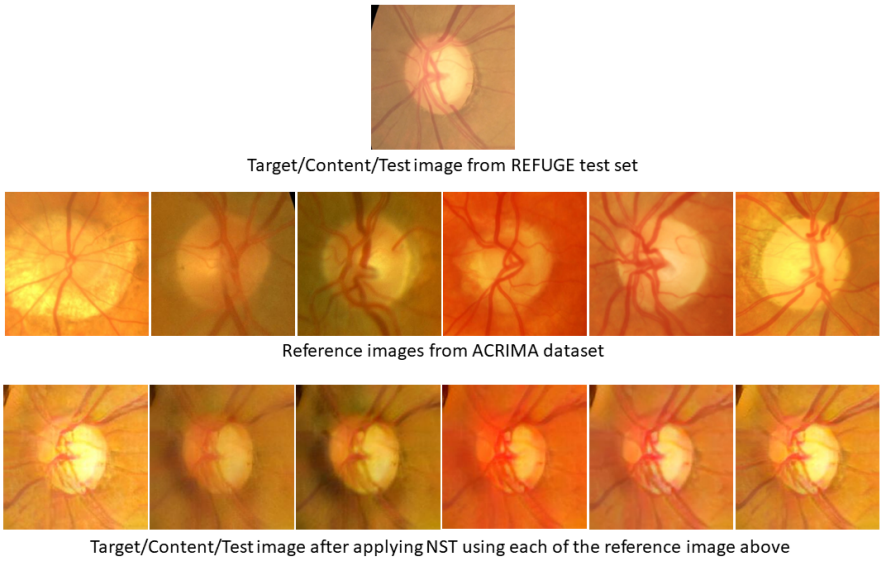


Figure 8.1: Example image of applying NST to test image in REFUGE dataset using reference images from ACRIMA dataset

its accuracy with the multi-image histogram matching algorithm introduced in Chapter 5. Next, we extended the experiments to see the performance of the NST method used with pre-trained machine learning-based glaucoma classifiers that are publicly available. Finally, we propose a framework fine-tuned for combined camera identification and glaucoma identification using NST as a test-time augmentation method.

8.2 Methodology

We applied NST as a test-time pre-processing step. REFUGE and RIMONE labelled glaucoma detection datasets drawn from three different models of fundus camera (1655 images in total) were used. REFUGE images were cropped around the optic nerve head to match the RIMONE images. In the RIMONE dataset, images are center-cropped around the optic nerve head (ONH) to focus on relevant anatomical features. To eliminate extraneous information and maintain consistency in image processing, we applied a similar center-cropping technique to the images in the REFUGE dataset.

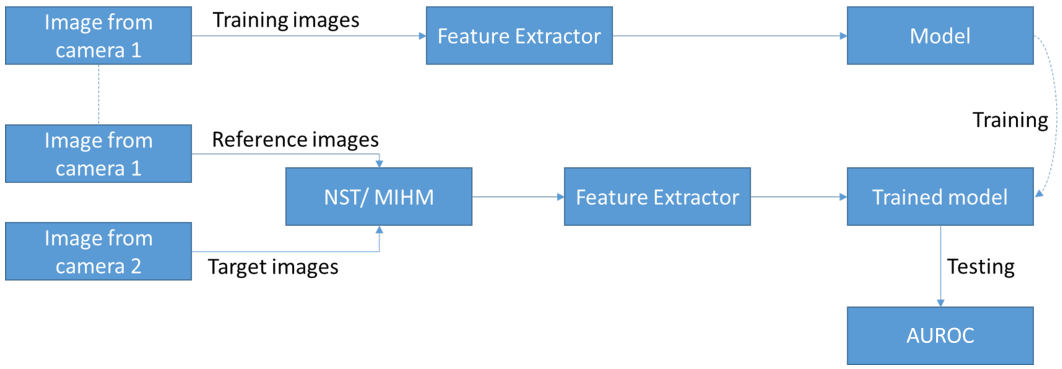


Figure 8.2: Application of neural style transferring for glaucoma label prediction on test images from camera 2 using reference images from training images acquired using camera 1

Then, NST was applied in the pre-processing step. Image features were extracted using ResNet101V2 and AlexNet pre-trained neural networks, and then glaucoma detection models based on random forests and XGBoost were trained. Furthermore, randomised multi-image histogram matching was applied separately to compare them. The experiment was conducted by assigning images from one camera as the training set and images from another as the target/test set, as shown in Figure 8.2 and using training images as reference images.

The experiment setup operates in two phases, namely, training and testing. First, features are extracted from the training images, and a classifier is trained using the features extracted. The second phase tests the classifier using the features of test images which are pre-processed. Both pre-processing methods require a reference image set, and our experiment uses training images to style/histogram match the test images. Reference images are chosen randomly. Finally, the test AUROC is observed.

Also, a second experiment was conducted by combining images from two cameras to form the training dataset. Images from the remaining camera were set as the test set. Random reference images are chosen equally from both subsets of the training dataset.

Table 8.1: Combination of feature extractors and classifiers in experiments

Number	Feature Extractor	Classifier
1	AlexNet	Random Forest
2	AlexNet	XGBoost
3	ResNet	Random Forest
4	ResNet	XGBoost

Table 8.2: Comparison of neural style transferring and multi-image histogram matching in test time - AUROC

Experiment No	Train	Test	Neural Style Transferring				Histogram Matching			
			1	2	3	4	1	2	3	4
(1)	REFD1	REFD2	**0.9314**	0.9213	0.8995	0.8216	0.6836	0.7013	0.7428	0.6946
	REFD1	RIM1	**0.8987**	0.7732	0.8046	0.7303	0.7129	0.7878	0.6961	0.6303
	REFD2	REFD1	**0.9193**	0.8442	0.7848	0.7831	0.7534	0.7922	0.7300	0.7361
	REFD2	RIM1	0.7117	0.6681	**0.8581**	0.8215	0.7317	0.6226	0.8488	0.8333
	RIM1	REFD1	0.8994	**0.9099**	0.8273	0.8089	0.6226	0.8093	0.7415	0.7313
	RIM1	REFD2	0.8521	0.8246	**0.9096**	0.8977	0.5661	0.7165	0.8585	0.8128
	Average		**0.8688**	0.8236	0.8473	0.8105	0.6784	0.7383	0.7696	0.7397
(2)	REFD2+RIM1	REFD1	**0.9174**	0.9139	0.8114	0.8213	0.7118	0.7649	0.7730	0.7483
	REFD1+RIM1	REFD2	0.8979	**0.9437**	0.8999	0.9001	0.6344	0.7450	0.8059	0.8060
	REFD1+REFD2	RIM1	0.8339	0.8240	**0.8548**	0.8358	0.6668	0.6324	0.8546	0.8123
	Average		0.8831	**0.8939**	0.8553	0.8524	0.6710	0.7141	0.8112	0.7889

We used two feature extractors and two classifiers in the experiments, as shown in Table 8.1. AlexNet and ResNet101V2 were the feature extractors, whereas random forest and XGBoost were the classifiers.

8.3 Results and discussion

For experiment 1, the highest average AUROC of 0.87 was obtained using the style transfer method, then a random forest classifier with features extracted using AlexNet. For histogram matching, the best average AUROC was 0.77.

In experiment 2, the highest average AUROC of 0.89 was obtained using the style transfer method, then an XGBoost classifier with features extracted using AlexNet. For histogram matching, the best average AUROC was 0.81.

Nevertheless, all the figures are above 0.80, which is promising and acceptable XGBoost as a good model [96].

Also, the style transfer-based pre-processed showed consistently improved accuracy compared to histogram matching in all four tested settings. However, the stylising process consumes more processing time than multi-image histogram matching. The stylisation process typically necessitates the processing of two images through a pretrained deep learning model, such as the VGG network. However, the multi-image histogram matching technique employs a simpler image processing step, which is repeated N times, resulting in a considerably shorter duration for the transformation.

8.4 Extended experiment 01

Given the promising results, we used NST to compare performance against publicly available pre-trained models. The aim was to investigate the reusability of readily available open-source models trained on a similar dataset (the training data is also free to use) without retraining those models on data from unseen cameras.

ACRIMA is a publicly available labelled dataset used to train five machine-learning models to identify glaucoma. The dataset contains 705 retinal fundus images, with 396 glaucoma images and 309 non-glaucoma images captured from a Topcon TRC retinal camera. The trained models are well-known CNNs, namely, Resnet50, VGG16, VGG19, InceptionV3 and Xception.

Initially, we tested the model accuracies against REFUGE and RIMONEr2 datasets with respect to original data from camera models to set up the lower bound baseline. Next, we performed test time stylisation of data using NST to match the training data. We conducted multiple experiments to select the best-generalised setup for testing.

To enhance the classification accuracy, it is crucial to select the reference image with careful consideration. The first experiment setup was stylising

a test image using the least correlated reference image in the full ACRIMA dataset from the same class as the test image. First, we conducted a comparative analysis of histograms between the reference image and the content image. We selected the reference image such that the difference between the histograms of the content image and the reference image was maximised.

However, searching in nearly half of the ACRIMA dataset per target image is time-consuming. Furthermore, the above stylisation is impossible in practical situations because of the lack of labelled target data unless an expert assigns an initial label before the test time. Hence, this setup acted as an upper-bound baseline.

Afterwards, we tested different methods that properly choose reference images for styling as shown in Algorithm 8.2. However, choosing the most suitable reference image for a certain target image is challenging.

8.4.1 Experimental setup

The experiment setup is mainly based on two modules, namely, selection and aggregation. The first module selects an appropriate reference image from ACRIMA. The second module aggregates multiple predictions after performing the stylisation, followed by predicting the label by an ACRIMA pre-trained model. The process diagram is shown in Figure 8.3. The blue, green and grey colour sections of the figure show the normal process flow, the high-level selection module and the high-level aggregation module, respectively.

The selection module may use a full reference dataset or a selected part of the dataset based on the clustering of reference images. The first approach takes more time to complete because it searches almost half of the reference data set. Hence, we chose the second approach to explore finding an accurate and efficient algorithm.

We clustered images of a certain class into 8 clusters using k-means clustering as shown in Table 8.3. Prior to that, image features are extracted using VGG16 ImageNet pre-trained feature extractor. The number of clusters was

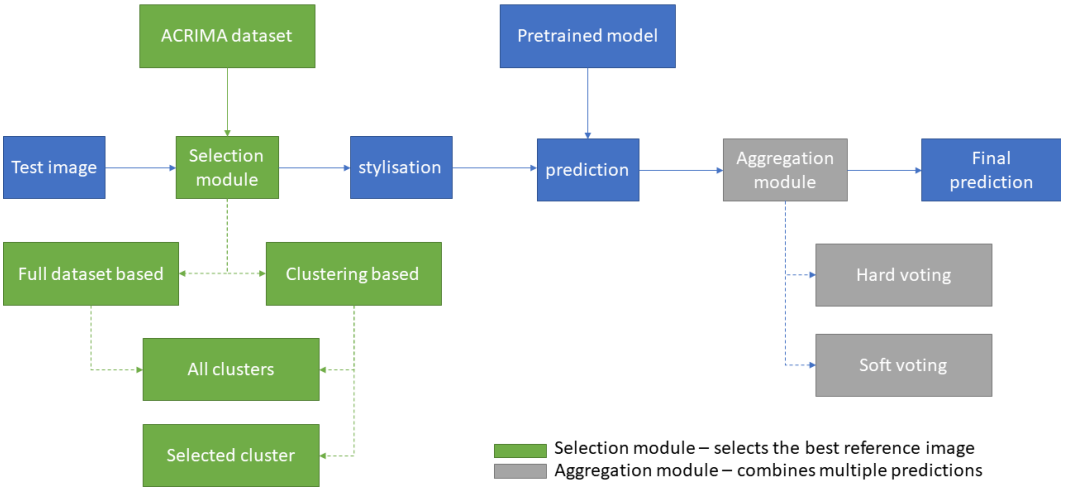


Figure 8.3: The process of applying NST in the experiments

chosen based on the elbow method [97] as the visualisation shown in Figure 8.4 where the elbow point is at 8 clusters (k = 8).

Afterwards, we chose cluster number 6 for future experiments as it had the most balanced proportions of images from each class. Image distribution in each cluster can be seen in Table 8.3. Furthermore, the accuracy given for cluster 6 was the highest when we initially tested the choice of the cluster by conducting class-wise stylising, which sets up an upper bound.

Each test image was styled using three random reference images from each class, resulting in six augmented images. Afterwards, we take predictions from all seven images, including the original image and max vote to get the final prediction. The experiment was repeated for each model separately.

8.4.2 Results and discussion

Testing with original test images, the average model accuracy varied among datasets from 0.8670 to 0.9348 AUROC, as shown in the rightmost column of Table 8.4. In contrast, the best stylising setup that takes reference images from the same class as the test image has increased the average variation to be

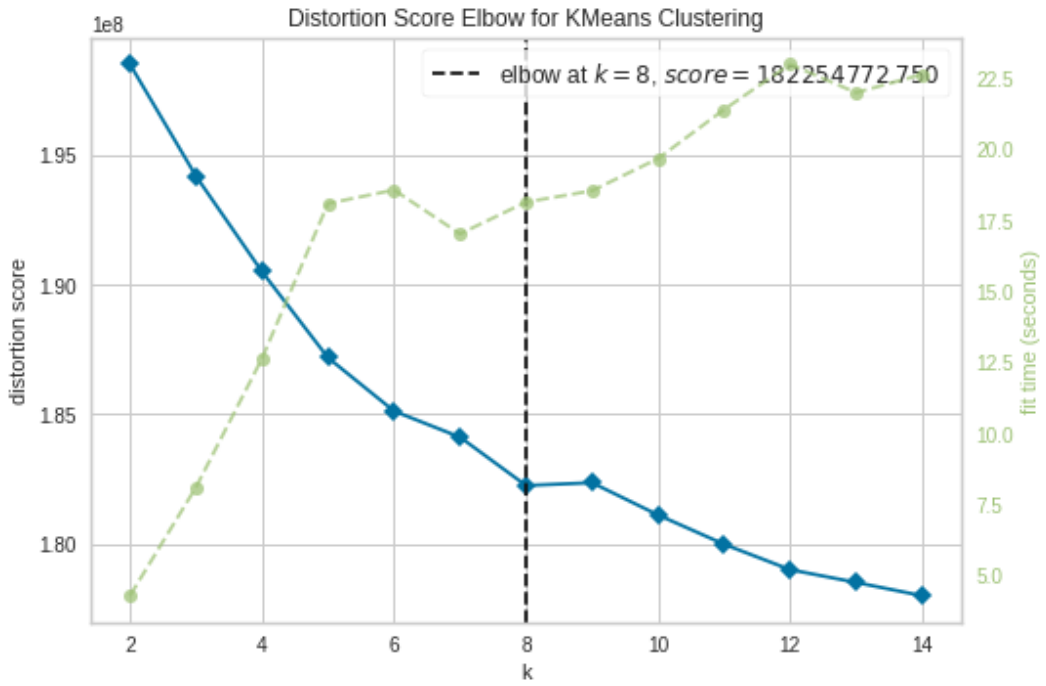


Figure 8.4: Visualisation of distortion score elbow for K-means clustering

Table 8.3: ACRIMA images in each cluster resulted from by K-Means clustering

Cluster	Non-glaucoma images	Glaucoma images	Total images
0	72	2	74
1	2	22	24
2	86	9	95
3	9	49	58
4	29	68	97
5	14	58	72
6	**74**	**77**	**151**
7	23	111	134
Total	309	396	705

Table 8.4: Test AUROC of ACRIMA model on original test data

Dataset	Xception	InceptionV3	VGG16	VGG19	ResNet50	Dataset average
RIMONE	0.9138	0.9335	0.9410	0.9497	0.9348	0.9346
REFUGED1	0.8010	0.8183	0.9371	0.9389	0.8344	0.8659
REFUGED2	0.8862	0.8938	0.8425	0.9157	0.9047	0.8886
Model average	0.8670	0.8819	0.9069	0.9348	0.8913	

Table 8.5: Test AUROC of ACRIMA model on styled test data using images from the same class of ACRIMA dataset

Dataset	Xception	InceptionV3	VGG16	VGG19	ResNet50	Dataset average
RIMONE	0.9620	0.9094	0.9788	0.9844	0.9259	0.9521
REFUGED1	0.9214	0.8684	0.9763	0.9898	0.9728	0.9457
REFUGED2	0.9563	0.9041	0.9997	1.0000	0.9483	0.9617
Model average	0.9466	0.8940	0.9849	0.9914	0.9490	

Table 8.6: Stylising of the test data using a random selection of reference images from ACRIMA dataset

Dataset	Xception	InceptionV3	VGG16	VGG19	ResNet50	Dataset average
RIMONE	0.8667	0.8526	0.7909	0.7032	0.8355	0.8098
REFUGED1	0.8210	0.7922	0.7186	0.7349	0.7706	0.7675
REFUGED2	0.8178	0.7579	0.7375	0.7934	0.8115	0.7836
Model average	0.8352	0.8009	0.7490	0.7438	0.8059	

Table 8.7: Class-wise stylising of the test data using reference images from the selected cluster (cluster number 6)

Dataset	Xception	InceptionV3	VGG16	VGG19	ResNet50	Dataset average
RIMONE	0.9616	0.9645	0.9588	0.9760	0.9772	0.9676
REFUGED1	0.9697	0.9621	0.9375	0.9776	0.9656	0.9625
REFUGED2	0.9596	0.9483	0.9531	0.9230	0.9390	0.9446
Model average	0.9636	0.9583	0.9498	0.9589	0.9606	

Table 8.8: Class independent stylising of the test data using reference images from the selected cluster (cluster number 6)

Dataset	Xception	InceptionV3	VGG16	VGG19	ResNet50	Dataset average
RIMONE	0.9573	0.9418	0.9556	0.9586	0.9590	0.9545
REFUGED1	0.8893	0.9065	0.9335	0.9529	0.8940	0.9152
REFUGED2	0.9461	0.9384	0.9130	0.9193	0.9248	0.9283
Model average	0.9309	0.9289	0.9340	0.9436	0.9260	

between 0.8940 and 0.9914 AUROC according to the results in the bottom row of Table 8.5. In contrast, stylising data using a random selection of reference images drastically reduces the performance, as depicted in Table 8.6.

Moreover, the best stylising method, which performs the class dependant stylising in the cluster-based setup, gives AUROC varied between 0.9498 and 0.9636 on average. It is comparable to the best results considering the processing time.

The final experiment that processes class-independent stylising shows a minimum average AUROC of 0.9260 and a maximum AUROC of 0.9436. Results improve the original prediction probability when augmented with stylised images. Furthermore, the suggested method approximates the predictions nearly as much as the cluster baseline.

8.5 Extended experiment 02

Previous experiment shows that each pre-trained model resulted in different AUROCs and that the best model is different for all the datasets. Hence, it was decided to find the best pre-trained model before predicting the test data. The aim of the following experiment was to predict using the best model without combining the results from all models, which may produce lower accuracy.

8.5.1 Experimental setup

The experimental setup consisted of two modules, namely, the device module and the glaucoma module that can recognise the camera model and the glaucoma class, respectively. Each module is a machine learning classifier model, where the device module is a multi-class classifier, and the glaucoma module is a binary classifier.

Device module

The device module can identify the camera model of a test image. We trained a YOLOv5 model using training images of four cameras: ACRIMA, RIMONEr2, REFUGED1 and REFUGED2. Eighty images were used to train the model with a split of 8:1:1 in train/validation/test split.

"You Only Look Once" (YOLO) [98] was originally created for object detection. However, it can be utilised for image classification tasks as the models are trained on the COCO [99] dataset. YOLOv5 also includes functionality for test time augmentation, which we utilised in our experiments too. The architecture of the model is shown in Figure 8.5. The diagram has three main parts: backbone, neck and head, comprising a cross-stage partial network, a path aggregation network and a YOLO layer, respectively.

The device module classifies images from any unknown device to the most relevant camera model. However, this module can be easily retrained by including novel camera models as YOLO provides faster training compared to other methods [98].

Glaucoma module

Any image through the system's device identification phase travels through the best model of the glaucoma module suggested by the device module.

The glaucoma module consists of five well-known CNNs ResNet50, VGG16, VGG19, InceptionV3 and Xception. We used the publicly available model weights pre-trained on the ACRIMA dataset. Each model can recognise if a

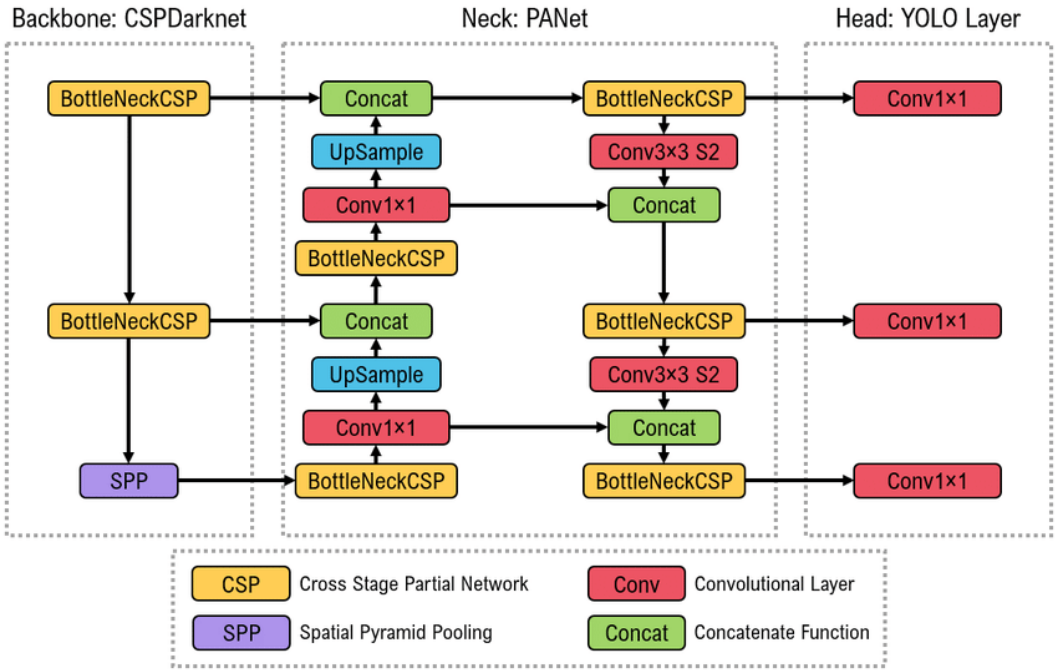


Figure 8.5: The architecture of the YOLOv5 model

test image is glaucomatous or not.

A complete system is proposed to integrate the above two modules, and a stylisation module resides between them. The high-level illustration of the overall system diagram is shown in Figure 8.6, where the prediction module utilises the glaucoma module.

The final prediction of the glaucoma status of an image is determined by the combined predictions of two augmented images and the original image itself. The stylisation module produces augmented images. The module stylises the original test image using two ACRIMA images from glaucoma and non-glaucoma classes. The reference ACRIMA images for each test image is chosen by comparing the histogram of each ACRIMA image against the histogram of the test image. We select the two most correlated ACRIMA images from both classes for stylising. Histogram comparison is lighter than pixel-based methods and can be easily measured using the correlation method.

The original test image and the stylised images are fed to the prediction module to generate the prediction probabilities of each image. Finally, the

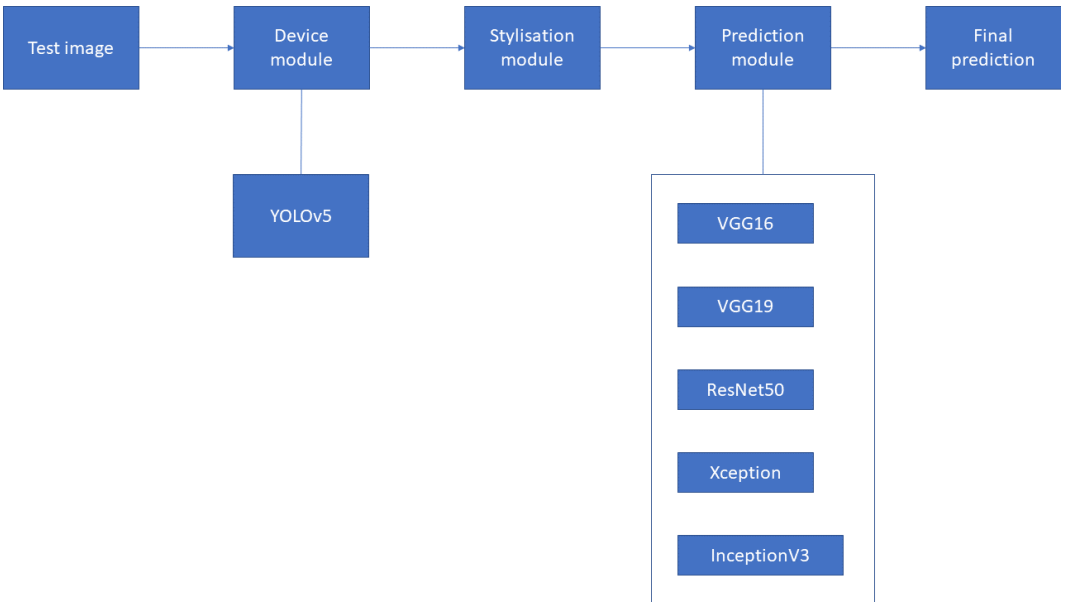


Figure 8.6: High level illustration of the overall proposed system

glaucoma class is assigned from the image shows the minimum entropy. The complete algorithm of the process is detailed in the Algorithm 8.2.

8.5.2 Results and discussion

We obtained the AUROCs of the best models selected by the device module and compared them with all other models under the same selection criteria. The results are shown in Table 8.9.

Table 8.9: Results of extended experiment 02
(Bold values in each row indicate the maximum AUROC value for each test dataset, highlighting the model that achieved this result.)

Dataset	Xception	InceptionV3	VGG16	VGG19	ResNet50
RIMONE	0.9179	**0.9512**	0.8662	0.8422	0.8770
REFUGED1	0.8820	0.8897	0.8533	**0.9467**	0.8094
REFUGED2	0.9036	0.8463	0.8542	0.8026	**0.9360**

The device module identified Inception3, VGG19 and ResNet50 as the best model for RIMONE, REFUGED1 and REFUGED2, respectively. The corresponding AUROCs are 0.9512, 0.9467 and 0.9360. The above values are

greater than the predictions for original images by each of the best models stated above. These models predicted 0.9335, 0.9389 and 0.9047 AUROCs for original images taken from the respective datasets. Therefore, adding a device identification module before the prediction module increases the predicting ability of the system by identifying the best model that gives better classification accuracy for a certain dataset when performing stylisation.

All experiments presented in this chapter were conducted using the Google Colaboratory environment, specifically with GPU support through the Pro version. The NVIDIA Tesla T4 GPU [100] was utilised, which is equipped with 2560 CUDA cores and 16 GB of GDDR6 memory. This GPU model also incorporates Turing Tensor Cores, enhancing its efficiency and suitability for AI-related workloads. On average, processing one dataset and achieving results in each experiment required approximately 1.5 hours.

Algorithm 8.1 Algorithm to select reference images, apply NST and aggregate predictions

procedure $f\_P(x, A, s, M)$ ▷ x= test image, A=initial reference set, s=sample size, M= model

$R = f\_A()$ Selecting best reference set using initial reference set A

$R' = f\_S(s)$ Choosing a sample of size s from the best reference set R

$x' = f\_R(x)$ Stylising x using reference set R‘

$y = f\_M(x')$ Predicting x‘ using model M

$y' = f\_Y(y)$ Aggregating predictions y

return y

procedure f_A1(A, k) ▷ A=initial reference set, k=number of clusters

$C = f\_K(A, k)$ Grouping A for k number of clusters

$c = f\_C_{min}(s)$ picking the cluster with the lowest entropy in the labels from C

return c

procedure f_A2(A, k) ▷ A=initial reference set, k=number of clusters

$C = f\_K(A, k)$ Grouping A for k number of clusters

$c = f\_C_{max}(s)$ picking the cluster with the lowest entropy in the labels from C

return c

procedure f_A3(A, k) ▷ A=initial reference set, k=number of clusters

$C = f\_K(A, k)$ Grouping A for k number of clusters

foreach $c \in \mathcal{C}$ **do**

$C'_c = f\_C_{center}(c)$ ▷ selecting a single image from c closest to centroid

end

return C‘ ▷ C‘ = set of selected images in c‘s

procedure f_A4(A, k) ▷ A=initial reference set, k=number of clusters

$C = f\_K(A, k)$ Grouping A for k number of clusters

foreach $c \in \mathcal{C}$ **do**

$C'_c = f\_C_{random}(c)$ ▷ selecting a random image from c

end

return C‘ ▷ C‘ = set of selected images in c‘s

=0

procedure f_Y1(P) ▷ P=array of prediction probabilities

$L = f\_P(A, k)$ ▷ assigning label (class giving max probability) to each element in array P

$l_0 = f\_L_0()$ ▷ counting zeros in array L

$l_1 = f\_L_1()$ ▷ counting ones in array L

$l = max(l_0, l_1)$

return l

procedure f_Y2(P) ▷ P=array of prediction probabilities

$L = f\_P(A, k)$ ▷ assigning label (class giving max probability) to each element in array P

$l_0 = sum(P_0)$ ▷ sum of probabilities of class zero in array P

$l_1 = sum(P_1)$ ▷ sum of probabilities of class one in array P

$l = max(l_0, l_1)$

return l

procedure f_Y3(P) ▷ P=array of prediction probabilities

$e = f\_E(P)$ ▷ calculating the entropy of each prediction of array P

$l_p = f\_L_{max}(e, P)$ ▷ assigning the prediction with maximum entropy as the class

return l_p

procedure f_Y4(P) ▷ P=array of prediction probabilities

$e = f\_E(P)$ ▷ calculating the entropy of each prediction of array P

$l_p = f\_L_{min}e, P$ ▷ assigning the prediction with minimum entropy as the class

return l_p

Algorithm 8.2 Generalised algorithm that predicts the glaucoma class of a test image based on camera model

procedure $f\_G(x, A, s, M)$ ▷ x= test image, A=initial reference set, s=sample size, M= model

$D = f\_C()$ ▷ Identifying the camera of the image x [Camera module]

$M = f\_M()$ ▷ Selecting the best model for D **if** $D! = 'ACRIMA'$ **then**

end

$S = f\_A(x)$ ▷ Augment x by stylising

$P = f\_P([x, S], M)$ ▷ Get predictions for S and x using M

$y = f\_Y(P)$ ▷ Aggregate the predictions **else**

end

$y = f\_P(x, M)$ ▷ Predict for x using M

return y

procedure $f\_Y(P)$ ▷ P=array of prediction probabilities

$e = f\_E(P)$ ▷ Calculating the entropy of each prediction of array P

$lp = f\_L(e, P)$ ▷ Assigning the prediction with minimum entropy as the class

return 1

procedure $f\_A(x)$ ▷ x=test image

$R_0 = f\_H(x, A_0)$ ▷ Select the best reference image by comparing the histograms of x and ACRIMA images of class 0

$R_1 = f\_H(x, A_1)$ ▷ Select the best reference image by comparing the histograms of x and ACRIMA images of class 1

$S = f\_S(R_0, R_1, x)$ ▷ Stylise x using the best reference images

return S

Chapter 9

Conclusions, Contributions and Future work

This chapter depicts the conclusions, contributions and future research directions raised from the study. We highlight the challenges and limitations of the research as well. Furthermore, in related studies, machine learning models—particularly those utilising deep learning techniques—have demonstrated promising outcomes in the classification of glaucoma. These models frequently achieve high accuracy rates, typically ranging from 85% to 95%, under controlled study conditions.

9.1 Summary

The initial research question focused on identifying the most effective machine learning-based feature extractors and determining the critical features for detecting glaucoma from retinal fundus images. To address this, we utilised the REFUGE dataset, along with 28 pre-trained machine learning models, and performed 10-fold cross-validation experiments using fully processed, cropped, and concatenated features from the images. The findings revealed that AlexNet demonstrated superior performance when employing concatenated features.

The second research question was directed towards improving the generalisation capabilities of machine learning models across different fundus cameras

in the classification of glaucoma. This was pursued by integrating image preprocessing techniques. In this study, shallow classifiers were trained using data exclusively from one model of fundus camera as the training set, while another model served as the testing set. This approach enabled the evaluation of classifier performance when preprocessing methods were applied at both the training and testing phases. The findings revealed that implementing histogram matching at test time, utilising training data, significantly enhanced the generalisability of the machine learning models.

The final research question aimed to explore the application of neural style transfer during the test phase on pretrained machine learning systems to enhance generalisation across different fundus cameras. For this purpose, publicly available pretrained glaucoma classifiers, originally developed using the ACRIMA dataset, were evaluated using images from the RIMONE and REFUGE datasets, with neural style transfer implemented during the testing phase. The findings indicated that neural style transfer substantially improves the generalisability of pretrained classifiers and minimises the necessity for retraining.

9.2 Conclusions

This section summarises the conclusions drawn from the Chapters 4 to 8.

First, we conducted a preliminary study to identify the problem domain. We compared 28 pre-trained models available in the Keras library for feature extraction in glaucoma detection in the REFUGE dataset. Early studies used pre-trained models for classification tasks of glaucoma by fine-tuning the base architecture or adding more complex layers than our study's tested approach.

Existing systems often require training with GPUs as they use much more advanced preprocessing along with diverse intensity normalisations and augmentation techniques. In contrast, our study shows that similar outcomes are desirable with minimum preprocessing and without training a complex neural

network for the REFUGE fundus images.

Another novelty of our research is the combination of features. Previous studies used either only the whole image features or extracted ROI only. However, we concatenated both the ONH and whole retinal fundus image features, creating an expanded feature set. This strategy demonstrates improved accuracy over other methods for the given data. It is because the combined feature set includes features of the entire image and twice the ONH area. In other words, the new feature set comprises emphasised ROI features and all additional features from whole and cropped versions of the retinal fundus image.

REFUGE dataset includes a validation dataset where the images come from a different camera. The system accuracy dropped when testing with the second dataset as they were not being used in the training process directly or indirectly. Therefore, we explored the possibilities and techniques for developing a more robust system that gives similar or greater accuracy when predicting labels for the images from cameras it was not trained for.

Accordingly, we have conducted several experiments to show that much simpler changes in image preprocessing can achieve generalisation on the images collected by different retinal fundus cameras. Therefore, we propose that a generalised system can be developed without consuming expensive resources. We used input standardisation, median filter and multi-image histogram matching in various combinations at the training and testing phases.

First, we changed the method introduced in [2] to standardise input images in YC_bC_r colour space, which exhibits improved average accuracy than the other methods. The modified input standardisation image preprocessing method is better with the median filter before training because the median filter method removes noise and preserves edges.

We introduced multi-image histogram matching that matches the histogram of the test image to multiple random reference images iteratively. It was applied during the testing phase using training images as reference data. The method outperformed conventional histogram matching in pilot experiments.

Another solution to the unseen domain problem in this research is the application of deep stacked transformations for augmenting training images. The idea is to change colour and spatial image information such that the resulting images are less specific to a certain camera model. However, selecting the best values of DST parameters is challenging. Hence, we used the surrogate optimisation technique to obtain optimal probabilities to apply each transformation keeping the magnitude the same. Finally, we compared classification accuracy with the optimised parameters. Although the results are below SOTA, the model performance is reasonable as we used a very constrained setup. Therefore, our solution can be useful in model training using image data from multiple devices to eliminate device dependency.

We designed a simple CNN called GlaucomaNet that generally outperforms the random forests and XGBoost classifiers. However, the XGBoost classifier works its best without any preprocessing step. Random forests, however, failed to achieve acceptable results. Moreover, the results show that simple GlaucomaNet can be sufficient for classifying glaucoma datasets while using deep stacked transforms and style transfer to increase generalisation under different camera settings.

We conducted another set of experiments that involved NST, where the test image was styled using an image of the training set. The purpose was to reduce the time to retrain a model for new retinal fundus images from another camera. The experiment setup used AlexNet and ResNet101V2 as feature extractors and random forest and XGBoost as classifiers, as previously found in this study. The results were compared against the multi-image histogram matching, which was better when trained and tested across the images from various cameras. Hence, NST can be used as a test time preprocessing method that reduces the requirement of retraining and fine-tuning a pre-trained model.

The style transfer-based preprocessing showed consistently improved accuracy compared to every other preprocessing method we tested because it uses a pre-trained CNN to transfer styles from one image to another. In doing

this, a defined loss function attempts to minimise the differences between the content image, the style reference image, and the generated image.

Later, we used NST to test pre-trained ACRIMA models against test images captured from various cameras different to the images the models were trained on. The results indicate that it is important that the best reference images are chosen from the reference dataset carefully. However, the automatic selection of the best reference image is challenging.

9.3 Contributions

One of the main objectives of this study is to compare existing pre-trained deep learning models as feature extractors in conjunction with standard computer vision template matching to identify the best feature extractors for glaucoma detection in retinal fundus images. The purpose was to feed the best features to train a shallow classifier. Another aspect of this study is that we analyse whole retinal fundus images and a region of an image, combined and separated. We use logistic regression and random forest as base classifiers in multiple cross-validation experiments to obtain accurate predictive performance estimates. They were chosen because they have been used in related research as conventional classifiers for a given feature set.

Another objective of this study is to compare the combinations of multiple existing preprocessing methods to examine how well they improve device domain generalisation for glaucoma detection in retinal fundus images. We considered median filtering, input standardisation and histogram matching with their combinations. Random forest classifiers have proven accurate with these types of images and image features in our previous experiments [76]. Hence, we use the random forest as the base classifier in all experiments to obtain accurate estimates of predictive performance.

As far as the authors know, this research is the first to consider the retinal fundus camera model variability issue using publicly available data. We

demonstrate that simple changes in image preprocessing can generalise machine learning models trained on the retinal fundus images created by different camera models. We altered one of the input standardisation methods to improve training accuracy. Although histogram equalisation has been a consideration in previous research, to the authors' knowledge, this is the first study to use histogram matching for a glaucoma detection task directly. As an extension to histogram matching, this research introduced a multi-image histogram matching algorithm to optimise model generalisation on test data. The algorithm is simple but effective as a preprocessing method because it is faster and uses fewer resources than NST, which will be discussed next. However, the algorithm requires a few training images to be used as reference images. Therefore, it will only sometimes be possible to use the proposed method concerning data privacy.

Deep stacked transformation application for the unseen domain problem in glaucoma classification was experimented with for the first time in this research. Also, the experiment design was complex and not seen in any of the previous studies. The complexity was introduced by choosing three different datasets as train, validation and test sets. Even though it contradicts the traditional machine learning requirement of validation and training sets to be similar, this setup can be identified as one suitable way to train and test a model for the unseen domain problem.

The final objective of the study was to examine the performance of applying data augmentation during the test time of a pre-trained machine learning model for its reusability. Input stylisation was the main consideration. NST has previously been used for fake image generation tasks, which performs well in related research [101]. This study is the first time the authors acknowledge using the technique as a test-time data augmentation method for glaucoma classification. The method has shown the potential of increasing the prediction accuracy than testing raw data.

9.4 Future work

At the conclusion of our experiments, we attempted to apply similar methodologies to a new private dataset comprised of fundus images captured using laser technology, in contrast to those obtained through conventional digital capturing techniques. The images derived from laser technology differed significantly from their digital counterparts. However, due to ethical considerations and constraints on timing, we were unable to proceed with this phase of the research. Nonetheless, the preliminary outlook on the potential outputs was positive, with anticipation for the application of the proposed solutions to the new data set. The processes of labeling the data and obtaining ethical approval proved time-consuming, which necessitated the suspension of this component of our study. However, future research could explore the feasibility of applying the proposed solutions to laser-based fundus images, thereby extending the scope of the current study and potentially validating the initial positive expectations.

One possible future research direction is to use similar methods as those described in this study for glaucoma stage classification as opposed to the binary classification (present/not present) problem, which currently is an understudied area of research [102]. Furthermore, the combined use of additional information, such as clinical features, making the problem multimodal is also a potential avenue for future research.

Optimising the preprocessing techniques' parameters will improve the results in future work. For example, additive and multiplicative constants of the standardisation method and the shape and size of the filter in median filtering can be optimally chosen. Optimising multi-image histogram matching iterations may help increase the test-time classification accuracy. Furthermore, it would be interesting to investigate the effect of the order of N images in contrast to the randomised strategy of multi-image histogram matching.

Furthermore, one can compare the results of surrogate optimisation with random augmentation (RandAug) and auto augmentation (AutoAug), another

potential policy optimisation technique to identify optimal DST parameters. Numerous alternative methods exist for identifying optimal learning parameters in classifiers. One such method is the Tree of Parzen Estimators, which Bergstra et al. suggest provides an efficient and effective means for selecting hyper-parameter settings [103].

Automatic choice of the best reference image in the NST technique is another important direction for further study. Choosing the best reference image is still challenging, and one can experiment with another method to compare images other than histogram comparison. Also, stylising may only be necessary for some predictions, and an algorithmic approach could be developed for deciding whether to style.

However, the translation of these results into clinical practice entails several challenges, such as obtaining regulatory approval, securing clinician acceptance, and ensuring continuous monitoring and validation across broader and more diverse populations.

The ability to accurately diagnose glaucoma from retinal fundus images, even when those images come from unseen domains, enhances the viability of telemedicine, especially in underserved or remote areas. This could significantly improve access to specialist healthcare services, reducing the need for patients to travel considerable distances for diagnosis, specially for differently abled people.

References

[1] Mamta Juneja, Niharika Thakur, Sarthak Thakur, Archit Uniyal, Anuj Wani, and Prashant Jindal, "Gc-net for classification of glaucoma in the retinal fundus image," *Machine Vision and Applications*, vol. 31, no. 5, pp. 1–18, 2020.

[2] Gwenolé Quellec, Mathieu Lamard, Pierre-Henri Conze, Pascale Massin, and Béatrice Cochener, "Automatic detection of rare pathologies in fundus photographs using few-shot learning," *Medical image analysis*, vol. 61, pp. 101660, 2020.

[3] Glaucoma Research Foundation, "Secondary glaucoma: Can diabetes cause glaucoma? - neovascular glaucoma," Online, Accessed: 2024.

[4] "Priority eye diseases - glaucoma," `https://www.who.int/blindness/causes/priority/en/index6.html`, Web site: World Health Organization.

[5] Glaucoma Patients, "Statistics," 2023, Accessed: 2023-04-20.

[6] Yih-Chung Tham, Xiang Li, Tien Y Wong, Harry A Quigley, Tin Aung, and Ching-Yu Cheng, "Global prevalence of glaucoma and projections of glaucoma burden through 2040: a systematic review and meta-analysis," *Ophthalmology*, vol. 121, no. 11, pp. 2081–2090, 2014.

[7] B Naveen Kumar, RP Chauhan, and Nidhi Dahiya, "Detection of glaucoma using image processing techniques: A review," in *2016 International Conference on Microelectronics, Computing and Communications (MicroCom)*. IEEE, 2016, pp. 1–6.

[8] Julian Zilly, Joachim M Buhmann, and Dwarikanath Mahapatra, "Glaucoma detection using entropy sampling and ensemble learning for automatic optic cup and disc segmentation," *Computerized Medical Imaging and Graphics*, vol. 55, pp. 28–41, 2017.

[9] Diego Torres Dias, Michele Ushida, Roberto Battistella, Syril Dorairaj, and Tiago Santos Prata, "Neurophthalmological conditions mimicking

glaucomatous optic neuropathy: analysis of the most common causes of misdiagnosis," *BMC Ophthalmology*, vol. 17, no. 1, pp. 2, 2017.

[10] Robert N Weinreb and Peng Tee Khaw, "Primary open-angle glaucoma," *The Lancet*, vol. 363, no. 9422, pp. 1711–1720, 2004.

[11] Abdullah Sarhan, Jon Rokne, and Reda Alhajj, "Glaucoma detection using image processing techniques: A literature review," *Computerized Medical Imaging and Graphics*, vol. 78, pp. 101657, 2019.

[12] U Rajendra Acharya, Sumeet Dua, Xian Du, and Chua Kuang Chua, "Automated diagnosis of glaucoma using texture and higher order spectra features," *IEEE Transactions on Information Technology in Biomedicine*, vol. 15, no. 3, pp. 449–455, 2011.

[13] Robert N Weinreb, Christopher K Leung, Jonathan G Crowston, Felipe A Medeiros, David S Friedman, Janey L Wiggs, and Keith R Martin, "Primary open-angle glaucoma," *Nature Reviews Disease Primers*, vol. 2, pp. 16067, 2016.

[14] Eric J. Topol, "High-performance medicine: the convergence of human and artificial intelligence," *Nature Medicine*, vol. 25, no. 1, pp. 44–56, 2019.

[15] Rajesh Kumar and Associates, "Navigating regulatory frameworks in ai health deployments," *International Journal of Medical Law and Ethics*, vol. 22, no. 3, pp. 300–320, 2022.

[16] Thomas White and Rebecca Harris, "Human-ai collaboration in healthcare: Enhancing diagnostic processes in ophthalmology," *Future of Healthcare Journal*, vol. 39, no. 1, pp. 45–59, 2022.

[17] Jeremy West, Dan Ventura, and Sean Warnick, "Spring research presentation: A theoretical foundation for inductive transfer," *Brigham Young University, College of Physical and Mathematical Sciences*, vol. 1, no. 08, 2007.

[18] Thommen George Karimpanal and Roland Bouffanais, "Self-organizing maps for storage and transfer of knowledge in reinforcement learning," *Adaptive Behavior*, vol. 27, no. 2, pp. 111–126, 2019.

[19] Yuan-Pin Lin and Tzyy-Ping Jung, "Improving eeg-based emotion classification using conditional transfer learning," *Frontiers in human neuroscience*, vol. 11, pp. 334, 2017.

[20] Baochen Sun, Jiashi Feng, and Kate Saenko, "Return of frustratingly easy domain adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2016, vol. 30.

[21] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané, "Concrete problems in ai safety," *arXiv preprint arXiv:1606.06565*, 2016.

[22] Ievgen Redko, Emilie Morvant, Amaury Habrard, Marc Sebban, and Younes Bennani, *Advances in domain adaptation theory*, Elsevier, 2019.

[23] Shiliang Sun, Honglei Shi, and Yuanbin Wu, "A survey of multi-source domain adaptation," *Information Fusion*, vol. 24, pp. 84–92, 2015.

[24] Zheyan Shen, Jiashuo Liu, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui, "Towards out-of-distribution generalization: A survey," *arXiv preprint arXiv:2108.13624*, 2021.

[25] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Wenjun Zeng, and Tao Qin, "Generalizing to unseen domains: A survey on domain generalization," *arXiv preprint arXiv:2103.03097*, 2021.

[26] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy, "Domain generalization: A survey," *arXiv preprint arXiv:2103.02503*, 2021.

[27] Jianhao Xiong, Andre Wang He, Meng Fu, Xinyue Hu, Yifan Zhang, Congxin Liu, Xin Zhao, and Zongyuan Ge, "Improve unseen domain generalization via enhanced local color transformation," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, Anne L. Martel, Purang Abolmaesumi, Danail Stoyanov, Diana Mateus, Maria A. Zuluaga, S. Kevin Zhou, Daniel Racoceanu, and Leo Joskowicz, Eds., Cham, 2020, pp. 433–443, Springer International Publishing.

[28] Patrick J. Saine and Marshall E. Tyler, *Ophthalmic Photography: Retinal photography, angiography, and electronic imaging*, Butterworth-Heinemann, Boston Mass., 2 edition, 2002.

[29] M. F. Armaly, "Optic Cup in Normal and Glaucomatous Eyes," *Investigative Ophthalmology & Visual Science*, vol. 9, no. 6, pp. 425–429, 06 1970.

[30] José Ignacio Orlando, Huazhu Fu, João Barbosa Breda, Karel van Keer, Deepti R. Bathula, Andrés Diaz-Pinto, Ruogu Fang, Pheng-Ann Heng, Jeyoung Kim, JoonHo Lee, Joonseok Lee, Xiaoxiao Li, Peng Liu, Shuai

Lu, Balamurali Murugesan, Valery Naranjo, Sai Samarth R. Phaye, Sharath M. Shankaranarayana, Apoorva Sikka, Jaemin Son, Anton van den Hengel, Shujun Wang, Junyan Wu, Zifeng Wu, Guanghui Xu, Yongli Xu, Pengshuai Yin, Fei Li, Xiulan Zhang, Yanwu Xu, and Hrvoje Bogunović, "Refuge challenge: A unified framework for evaluating automated methods for glaucoma assessment from fundus photographs," *Medical Image Analysis*, vol. 59, pp. 1361–8415, 2020.

[31] Francisco José Fumero Batista, Tinguaro Diaz-Aleman, Jose Sigut, Silvia Alayon, Rafael Arnay, and Denisse Angel-Pereira, "Rim-one dl: A unified retinal image database for assessing glaucoma using deep learning," *Image Analysis & Stereology*, vol. 39, no. 3, pp. 161–167, 2020.

[32] Hansi N Gunasinghe, James McKelvie, Abigail Koay, and Michael Mayo, "Automated detection of glaucoma from retinal fundus images using a variety of fundus cameras," in *Clinical and Experimental Ophthalmology*. WILEY 111 River Street, Hoboken 07030-5774, NJ USA, 2022, vol. 49, pp. 911–911.

[33] Andrew Chen, Suveera Dang, Mina M. Chung, Rajeev S. Ramchandran, Angela P. Bessette, David A. DiLoreto, David M. Kleinman, Jayanth Sridhar, Charles C. Wykoff, and Ajay E. Kuriyan, "Quantitative comparison of fundus images by 2 ultra-widefield fundus cameras," *Ophthalmology Retina*, vol. 5, no. 5, pp. 450–457, 2021.

[34] Jili Chen, "Comparison of the performance of four fundus cameras in clinical practice," *Investigative Ophthalmology & Visual Science*, vol. 60, no. 9, pp. 6121–6121, Jul 2019.

[35] Sibghatullah I Khan, Shruti Bhargava Choubey, Abhishek Choubey, Abhishek Bhatt, Pandya Vyomal Naishadhkumar, and Mohammed Mahaboob Basha, "Automated glaucoma detection from fundus images using wavelet-based denoising and machine learning," *Concurrent Engineering*, vol. 30, no. 1, pp. 103–115, 2022.

[36] José Denes Lima Araújo, Johnatan Carvalho Souza, Otilio Paulo Silva Neto, Jefferson Alves de Sousa, João Dallyson Sousa de Almeida, Anselmo Cardoso de Paiva, Aristófanes Corrêa Silva, Geraldo Braz Junior, and Marcelo Gattass, "Glaucoma diagnosis in fundus eye images using diversity indexes," *Multimedia Tools and Applications*, vol. 78, no. 10, pp. 12987–13004, 2019.

[37] Tai Ho Phuong Thanh, Tien Pham Thi Thuy, Truong Ngo Hieu, and Minh Son Nguyen, "A real-time classification of glaucoma from retinal

fundus images using ai technology," in *2020 International Conference on Advanced Computing and Applications (ACOMP)*. IEEE, 2020, pp. 114–121.

[38] Marriam Nawaz, Tahira Nazir, Ali Javed, Usman Tariq, Hwan-Seung Yong, Muhammad Attique Khan, and Jaehyuk Cha, "An efficient deep learning approach to automatic glaucoma detection using optic disc and optic cup localization," *Sensors*, vol. 22, no. 2, pp. 434, 2022.

[39] MB Sudhan, M Sinthuja, S Pravinth Raja, J Amutharaj, G Charlyn Pushpa Latha, S Sheeba Rachel, T Anitha, T Rajendran, and Yosef Asrat Waji, "Segmentation and classification of glaucoma using u-net with deep learning model," *Journal of Healthcare Engineering*, vol. 2022, 2022.

[40] Shailesh Kumar and Basant Kumar, "Automatic early glaucoma detection by extracting parapapillary atrophy and optic disc from fundus image using svm," *Multimedia Tools and Applications*, pp. 1–23, 2022.

[41] Poonguzhali Elangovan, Malaya Kumar Nath, et al., "Detection of glaucoma from fundus image using pre-trained densenet201 model," *Indian Journal of Radio & Space Physics (IJRSP)*, vol. 50, no. 1, pp. 33–39, 2022.

[42] Alan Carlos de Moura Lima, Lucas Bezerra Maia, Roberto Matheus Pinheiro Pereira, Geraldo Braz Junior, Joao Dallyson Sousa de Almeida, and Anselmo Cardoso de Paiva, "Glaucoma diagnosis over eye fundus image through deep features," in *2018 25th International Conference on Systems, Signals and Image Processing (IWSSIP)*. IEEE, 2018, pp. 1–4.

[43] Francisco Fumero, Silvia Alayón, José L Sanchez, Jose Sigut, and M Gonzalez-Hernandez, "Rim-one: An open retinal image database for optic nerve evaluation," in *2011 24th International Symposium on Computer-based Medical Systems (CBMS)*. 2011, pp. 1–6, IEEE.

[44] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 770–778, IEEE.

[45] Christian Szegedy, S. Ioffe, V. Vanhoucke, and Alexander Amir Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Association for the Advancement of Artificial Intelligence*, 2017.

[46] Deepak Ranjan Nayak, Dibyasundar Das, Banshidhar Majhi, Sulatha V Bhandary, and U Rajendra Acharya, "Ecnet: An evolutionary convolutional network for automated glaucoma detection using fundus images," *Biomedical Signal Processing and Control*, vol. 67, pp. 102559, 2021.

[47] Masakazu Hirota, Atsushi Mizota, Tatsuya Mimura, Takao Hayashi, Junichi Kotoku, Tomohiro Sawa, and Kenji Inoue, "Effect of color information on the diagnostic performance of glaucoma in deep learning using few fundus images," *International Ophthalmology*, vol. 40, no. 11, pp. 3013–3022, 2020.

[48] S Yedukrishnan, V Geetha, and V Jalaja Jayalakshmi, "Glaucoma detection in fundus image using cnn," in *2021 International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA)*. 2021, pp. 1–4, IEEE.

[49] Jayanthi Sivaswamy, SR Krishnadas, Gopal Datt Joshi, Madhulika Jain, and A Ujjwaft Syed Tabish, "Drishti-gs: Retinal image dataset for optic nerve head (onh) segmentation," in *2014 IEEE 11th international symposium on biomedical imaging (ISBI)*. IEEE, 2014, pp. 53–56.

[50] Saumya Borwankar, Raima Sen, and Bhavin Kakani, "Improved glaucoma diagnosis using deep learning," in *2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*. IEEE, 2020, pp. 1–4.

[51] Ryo Asaoka, Masaki Tanito, Naoto Shibata, Keita Mitsuhashi, Kenichi Nakahara, Yuri Fujino, Masato Matsuura, Hiroshi Murata, Kana Tokumo, and Yoshiaki Kiuchi, "Validation of a deep learning model to screen for glaucoma using images from different fundus cameras and data augmentation," *Ophthalmology Glaucoma*, vol. 2, no. 4, pp. 224–231, 2019.

[52] Ayesha Shoukat, Shahzad Akbar, and Khadija Safdar, "A deep learning-based automatic method for early detection of the glaucoma using fundus images," in *2021 International Conference on Innovative Computing (ICIC)*. IEEE, 2021, pp. 1–6.

[53] Ayesha Shoukat, Shahzad Akbar, Syed Al E Hassan, Amjad Rehman, and Noor Ayesha, "An automated deep learning approach to diagnose glaucoma using retinal fundus images," in *2021 International Conference on Frontiers of Information Technology (FIT)*. IEEE, 2021, pp. 120–125.

[54] C Sharmila and N Shanthi, "Retinal image analysis for glaucoma detection using transfer learning," in *Advances in Electrical and Computer Technologies*, pp. 235–244. Springer, 2021.

[55] Manop Phankokkruad, "Evaluation of deep transfer learning models in glaucoma detection for clinical application," in *2021 4th International Conference on Information and Communications Technology (ICOIACT)*. IEEE, 2021, pp. 114–118.

[56] Sertan Serte and Ali Serener, "A generalized deep learning model for glaucoma detection," in *2019 3rd International symposium on multidisciplinary studies and innovative technologies (ISMSIT)*. IEEE, 2019, pp. 1–5.

[57] Arkaja Saxena, Abhilasha Vyas, Lokesh Parashar, and Upendra Singh, "A glaucoma detection using convolutional neural network," in *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*. IEEE, 2020, pp. 815–820.

[58] Xi Xu, Yu Guan, Jianqiang Li, Zerui Ma, Li Zhang, and Li Li, "Automatic glaucoma detection based on transfer induced attention network," *BioMedical Engineering OnLine*, vol. 20, no. 1, pp. 1–19, 2021.

[59] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton, "Dynamic routing between capsules," *Advances in neural information processing systems*, vol. 30, 2017.

[60] Patrick Ryan Sales dos Santos, Vitória de Carvalho Brito, Antonio Oseas de Carvalho Filho, Flávio Henrique Duarte de Araújo, Ricardo de Andrade Lira Rabêlo, and Mano Joseph Mathew, "A capsule network-based for identification of glaucoma in retinal images," in *2020 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2020, pp. 1–6.

[61] Poonguzhali Elangovan and Malaya Kumar Nath, "Glaucoma assessment from color fundus images using convolutional neural network," *International Journal of Imaging Systems and Technology*, vol. 31, no. 2, pp. 955–971, 2021.

[62] Javier Civit-Masot, Manuel J Domínguez-Morales, Saturnino Vicente-Díaz, and Anton Civit, "Dual machine-learning system to aid glaucoma diagnosis using disc and cup feature extraction," *IEEE Access*, vol. 8, pp. 127519–127529, 2020.

[63] S Prashanth, HC Navyashree, G Vardhini, and R Nagesh, "Early and efficient detection of glaucoma using image processing and deep learning," *Perspectives in Communication, Embedded-systems and Signal-processing-PiCES*, vol. 4, no. 9, pp. 222–231, 2020.

[64] N Deepa, S Esakkirajan, B Keerthiveena, and S Bala Dhanalakshmi, "Automatic diagnosis of glaucoma using ensemble based deep learning model," in *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*. IEEE, 2021, vol. 1, pp. 536–541.

[65] S Choi, "sjchoi86-hrf database," .

[66] Andres Diaz-Pinto, Sandra Morales, Valery Naranjo, Thomas Köhler, Jose M Mossi, and Amparo Navea, "Cnns for automatic glaucoma assessment using fundus images: an extensive validation," *BioMedical Engineering OnLine*, vol. 18, no. 1, pp. 29, 2019.

[67] Ali Serener and Sertan Serte, "Glaucoma classification via deep learning ensembles," in *2021 International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*. IEEE, 2021, pp. 1–5.

[68] Jin Mo Ahn, Sangsoo Kim, Kwang-Sung Ahn, Sung-Hoon Cho, Kwan Bok Lee, and Ungsoo Samuel Kim, "A deep learning model for the detection of both advanced and early glaucoma using fundus photography," *PloS One*, vol. 13, no. 11, pp. e0207982, 2018.

[69] Donald C. Hood and Ali S. Raza, "Detecting glaucoma with a portable brain imaging device," *Progress in Retinal and Eye Research*, vol. 31, no. 5, pp. 400–425, 2012.

[70] Susanta Sarangi, Md Sahidullah, and Goutam Saha, "Optimization of data-driven filterbank for automatic speaker verification," *Digital Signal Processing*, vol. 104, pp. 102795, 2020.

[71] Ethem Alpaydin, *Introduction to machine learning*, MIT press, 2020.

[72] François Chollet, James Bradbury, Jonathon Shlens, Charles Correa, Vipul Rao, Pranab Kishore, Alexander Kopylov, Hydai Phan, Andrew Gibson, Pavel Weaver, Danijar Hafner, Shanqing Cai, Cedric Danier, Tomasz Kalinowski, and Jasper van der Jeugt, "Keras," `https://keras.io`, 2015.

[73] Leo Breiman, "Random forests," in *Machine Learning*, 2001, pp. 5–32.

[74] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[75] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[76] Hansi Gunasinghe, James McKelvie, Abigail Koay, and Michael Mayo, "Comparison of pretrained feature extractors for glaucoma detection," in *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, 2021, pp. 390–394.

[77] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, "Imagenet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, may 2017.

[78] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[79] Keith A Goatman, A David Whitwam, A Manivannan, John A Olson, and Peter F Sharp, "Colour normalisation of retinal images," in *Proceedings of medical image understanding and analysis*. The University of Sheffield United Kingdom, 2003, pp. 49–52.

[80] R.C. Gonzalez and R.E. Woods, *Digital Image Processing*, Pearson, 2018.

[81] Ayesha Shoukat, Shahzad Akbar, and Khadija Safdar, "A deep learning-based automatic method for early detection of the glaucoma using fundus images," in *2021 International Conference on Innovative Computing (ICIC)*, 2021, pp. 1–6.

[82] T. Huang, G. Yang, and G. Tang, "A fast two-dimensional median filtering algorithm," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 27, no. 1, pp. 13–18, 1979.

[83] Hansi Gunasinghe, James McKelvie, Abigail Koay, and Michael Mayo, "Domain generalisation for glaucoma detection in retinal images from unseen fundus cameras," in *Intelligent Information and Database Systems*, Ngoc Thanh Nguyen, Tien Khoa Tran, Ualsher Tukayev, Tzung-Pei Hong, Bogdan Trawiński, and Edward Szczerbicki, Eds., Cham, 2022, pp. 421–433, Springer Nature Switzerland.

[84] Ling Zhang, Xiaosong Wang, Dong Yang, Thomas Sanford, Stephanie Harmon, Baris Turkbey, Bradford J Wood, Holger Roth, Andriy Myronenko, Daguang Xu, et al., "Generalizing deep learning for medical image segmentation to unseen domains via deep stacked transformation," *IEEE transactions on medical imaging*, vol. 39, no. 7, pp. 2531–2540, 2020.

[85] David Eriksson, David Bindel, and Christine A Shoemaker, "pysot and poap: An event-driven asynchronous framework for surrogate optimization," *arXiv preprint arXiv:1908.00420*, 2019.

[86] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, Software available from tensorflow.org.

[87] E. Khvedchenya V. I. Iglovikov A. Buslaev, A. Parinov and A. A. Kalinin, "Albumentations: fast and flexible image augmentations," *ArXiv e-prints*, 2018.

[88] Connor Shorten and Taghi M Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of big data*, vol. 6, no. 1, pp. 1–48, 2019.

[89] NVIDIA Corporation, "NVIDIA GeForce RTX 3080 GPU," `https://www.nvidia.com/en-us/geforce/graphics-cards/30-series/rtx-3080/`, 2021.

[90] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[91] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur, "A learned representation for artistic style," *arXiv preprint arXiv:1610.07629*, 2016.

[92] Ziyang Chen, Yongsheng Pan, and Yong Xia, "Reconstruction-driven dynamic refinement based unsupervised domain adaptation for joint optic disc and cup segmentation," *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 7, pp. 3537–3548, 2023.

[93] Yalan Ye, Ziqi Liu, Yangwuyong Zhang, Jingjing Li, and Hengtao Shen, "Alleviating style sensitivity then adapting: Source-free domain adaptation for medical image segmentation," in *Proceedings of the 30th ACM International Conference on Multimedia*, New York, NY, USA, 2022, MM '22, p. 1935–1944, Association for Computing Machinery.

[94] Yann LeCun, Bernhard Boser, John Denker, Donnie Henderson, Wayne Hubbard, and Lawrence Jackel, "Handwritten digit recognition with a back-propagation network," *Advances in neural information processing systems*, vol. 2, 1989.

[95] Manuel Fernández-Delgado, Eva Cernadas, Senén Barro, and Dinani Amorim, "Do we need hundreds of classifiers to solve real world classification problems?," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3133–3181, 2014.

[96] Judith Kouassi Nzoughet, Khadidja Guehlouz, Stéphanie Leruez, Philippe Gohier, Cinzia Bocca, Jeanne Muller, Odile Blanchet, Dominique Bonneau, Gilles Simard, Dan Milea, Vincent Procaccio, Guy Lenaers, Juan M. Chao de la Barca, and Pascal Reynier, "A data mining metabolomics exploration of glaucoma," *Metabolites*, vol. 10, no. 2, 2020.

[97] Chunhui Yuan and Haitao Yang, "Research on k-value selection method of k-means clustering algorithm," *J*, vol. 2, no. 2, pp. 226–235, 2019.

[98] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[99] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision – ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, Eds., Cham, 2014, pp. 740–755, Springer International Publishing.

[100] NVIDIA Corporation, "NVIDIA Tesla T4 GPU," `https://www.nvidia.com/en-us/data-center/tesla-t4/`, 2022, [Online; accessed 20-11-2022].

[101] M.D. Toth and A. Kiss, "Retinal blood vessel segmentation on style-augmented images," *Studia Universitatis Babeș-Bolyai Informatica*, vol. 66, no. 1, pp. 74–85, 2021.

[102] Shilpa Sameer Kanse and Dinkar Manik Yadav, "Retinal fundus image for glaucoma detection: A review and study," *Journal of Intelligent Systems*, vol. 28, no. 1, pp. 43–56, 2019.

[103] James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl, "Algorithms for hyper-parameter optimization," in *Advances in Neural Information Processing Systems*, J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K.Q. Weinberger, Eds. 2011, vol. 24, Curran Associates, Inc.

Appendix A: List of retinal fundus camera models, manufacturers and publicly available datasets

Manufacturer	Model	Datasets	Image count	Glaucoma	Non Glaucoma	full/cropped around ONH
Canon	CR-2	REFUGE02	800	80	720	full
Canon	CR-2 AF					
Canon	CR-2 Plus AF					
Canon	CX-1					
Canon	CF-60UVi	High-Resolution Fundus Quality Assessment	30	15	15	full
EasyScan						
Forus Health	CLASSIC					
Kowa Optimed	Nonmyd WX-3D					
Kowa Optimed	KOWA VX-10α					
Kowa Optimed	VX-20					
NIDEK	AFC-210	RIM-ONE V2	455	200	255	cropped
NIDEK	AFC-330	Glaucoma Fundus	1542	756	786	cropped
NIDEK	Mirante(?)					
NIDEK	Retina Scan Duo™					
NIDEK	NM-200D					
S4Optik	Cobra					
Topcon	TRC-50DX					
Topcon	TRC-NW8					
Topcon	TRC-NW8F					
Topcon	TRC retina camera	ACRIMA	705	396	309	cropped
Volk	Pictor Plus					
Volk	iNview					
ZEISS	CIRRUS™ photo 600					
ZEISS	CIRRUS™ photo 800					
ZEISS	FF 450plus					
ZEISS	VISUCAM 200					
ZEISS	VISUCAM 224					
ZEISS	VISUCAM 500	REFUGE01	400	40	360	full
ZEISS		INSPIRE	40	40		
ZEISS	VISUSCOUT 100					
ZEISS	Visucam NM/FA	Drishti-GS1	101	70	31	full
Device Not Recorded		Retina	601	101	500	full

Appendix B: the table of t-values by degrees of freedom (dof) and desired probability that the Null should be rejected

***t* Table**

cum. prob	$t_{.50}$	$t_{.75}$	$t_{.80}$	$t_{.85}$	$t_{.90}$	$t_{.95}$	$t_{.975}$	$t_{.99}$	$t_{.995}$	$t_{.999}$	$t_{.9995}$
one-tail	**0.50**	**0.25**	**0.20**	**0.15**	**0.10**	**0.05**	**0.025**	**0.01**	**0.005**	**0.001**	**0.0005**
two-tails	**1.00**	**0.50**	**0.40**	**0.30**	**0.20**	**0.10**	**0.05**	**0.02**	**0.01**	**0.002**	**0.001**
df											
1	0.000	1.000	1.376	1.963	3.078	6.314	12.71	31.82	63.66	318.31	636.62
2	0.000	0.816	1.061	1.386	1.886	2.920	4.303	6.965	9.925	22.327	31.599
3	0.000	0.765	0.978	1.250	1.638	2.353	3.182	4.541	5.841	10.215	12.924
4	0.000	0.741	0.941	1.190	1.533	2.132	2.776	3.747	4.604	7.173	8.610
5	0.000	0.727	0.920	1.156	1.476	2.015	2.571	3.365	4.032	5.893	6.869
6	0.000	0.718	0.906	1.134	1.440	1.943	2.447	3.143	3.707	5.208	5.959
7	0.000	0.711	0.896	1.119	1.415	1.895	2.365	2.998	3.499	4.785	5.408
8	0.000	0.706	0.889	1.108	1.397	1.860	2.306	2.896	3.355	4.501	5.041
9	0.000	0.703	0.883	1.100	1.383	1.833	2.262	2.821	3.250	4.297	4.781
10	0.000	0.700	0.879	1.093	1.372	1.812	2.228	2.764	3.169	4.144	4.587
11	0.000	0.697	0.876	1.088	1.363	1.796	2.201	2.718	3.106	4.025	4.437
12	0.000	0.695	0.873	1.083	1.356	1.782	2.179	2.681	3.055	3.930	4.318
13	0.000	0.694	0.870	1.079	1.350	1.771	2.160	2.650	3.012	3.852	4.221
14	0.000	0.692	0.868	1.076	1.345	1.761	2.145	2.624	2.977	3.787	4.140
15	0.000	0.691	0.866	1.074	1.341	1.753	2.131	2.602	2.947	3.733	4.073
16	0.000	0.690	0.865	1.071	1.337	1.746	2.120	2.583	2.921	3.686	4.015
17	0.000	0.689	0.863	1.069	1.333	1.740	2.110	2.567	2.898	3.646	3.965
18	0.000	0.688	0.862	1.067	1.330	1.734	2.101	2.552	2.878	3.610	3.922
19	0.000	0.688	0.861	1.066	1.328	1.729	2.093	2.539	2.861	3.579	3.883
20	0.000	0.687	0.860	1.064	1.325	1.725	2.086	2.528	2.845	3.552	3.850
21	0.000	0.686	0.859	1.063	1.323	1.721	2.080	2.518	2.831	3.527	3.819
22	0.000	0.686	0.858	1.061	1.321	1.717	2.074	2.508	2.819	3.505	3.792
23	0.000	0.685	0.858	1.060	1.319	1.714	2.069	2.500	2.807	3.485	3.768
24	0.000	0.685	0.857	1.059	1.318	1.711	2.064	2.492	2.797	3.467	3.745
25	0.000	0.684	0.856	1.058	1.316	1.708	2.060	2.485	2.787	3.450	3.725
26	0.000	0.684	0.856	1.058	1.315	1.706	2.056	2.479	2.779	3.435	3.707
27	0.000	0.684	0.855	1.057	1.314	1.703	2.052	2.473	2.771	3.421	3.690
28	0.000	0.683	0.855	1.056	1.313	1.701	2.048	2.467	2.763	3.408	3.674
29	0.000	0.683	0.854	1.055	1.311	1.699	2.045	2.462	2.756	3.396	3.659
30	0.000	0.683	0.854	1.055	1.310	1.697	2.042	2.457	2.750	3.385	3.646
40	0.000	0.681	0.851	1.050	1.303	1.684	2.021	2.423	2.704	3.307	3.551
60	0.000	0.679	0.848	1.045	1.296	1.671	2.000	2.390	2.660	3.232	3.460
80	0.000	0.678	0.846	1.043	1.292	1.664	1.990	2.374	2.639	3.195	3.416
100	0.000	0.677	0.845	1.042	1.290	1.660	1.984	2.364	2.626	3.174	3.390
1000	0.000	0.675	0.842	1.037	1.282	1.646	1.962	2.330	2.581	3.098	3.300
z	0.000	0.674	0.842	1.036	1.282	1.645	1.960	2.326	2.576	3.090	3.291
	0%	50%	60%	70%	80%	90%	95%	98%	99%	99.8%	99.9%
	Confidence Level										

t-table.xls 7/14/2007