DOMAIN ADAPTATION TECHNIQUES FOR REAL-WORLD GLAUCOMA SCREENING

A PROJECT REPORT

Submitted by

PS Karthikeyan(953621104021)
S Santhosh(953621104035)

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING in

COMPUTER SCIENCE AND ENGINEEERING

RAMCO INSTITUTE OF TECHNOLOGY,

RAJAPALAYAM

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2025

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report "**DOMAIN ADAPTATION TECHNIQUES FOR REAL-WORLD GLAUCOMA SCREENING**" is the Bonafide work of "**SANTHOSH S (953621104035),PS KARTHIKEYAN (953621104021)**" who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr. K. Vijayalakshmi M.E., Ph.D.	Mr.R.Balamurugan B.E., M.E.
HEAD OF THE DEPARTMENT	**SUPERVISOR**
Professor & Head	Assistant Professor
Department of Computer Science and Engineering	Department of Computer Science and Engineering
Ramco Institute of Technology	Ramco Institute of Technology
North Venganallur Village	North Venganallur Village
Rajapalayam – 626117	Rajapalayam – 626117

Submitted for the Project Viva-voce held at Ramco Institute of Technology, Rajapalayam on _____

INTERNAL EXAMINER **EXTERNALEXAMINER**

ACKNOWLEDGEMENT

We are greatly indebted to our beloved Chairman **Shri. P. R. VENKETRAMA RAJA** for providing excellent facilities and encouragement to enhance our skills with world-class education by pursuing online courses which helped us greatly to carry out the project work successfully and sincerely thank our Principal **Dr. L. GANESAN**, for his continuous support and advice.

We would like to express our sincere gratitude to **Dr. K. VIJAYALAKSHMI**, Professor and Head, Department of Computer Science and Engineering for her support and encouragement for the completion of this project work.

We are grateful to our project guide **Mr.R.BALAMURUGAN,** Assistant Professor**,** Department of Computer Science and Engineering for her insightful comments and valuable suggestions which helped us to complete this project work successfully.

We would like to thank our project coordinators**, Mrs. S. MANJULA** Assistant Professor (Senior Grade), Department of Computer Science and Engineering for their support and suggestions. Our sincere thanks go to revered faculty members and Technical Staff members for their help during the course of this project work.

Last but not least, we extend our indebtedness towards our beloved parents for their support which made this project a successful one

PS.KARTHIKEYAN(953621104021),

S.SANTHOSH(953621104035)

ABSTRACT

Glaucoma, one of the leading causes of permanent blindness worldwide, often progresses silently until it reaches an advanced stage. Early detection is essential to preventing vision loss, and color fundus photography is a widely used non-invasive screening method. However, the significant variation in image quality across different screening environments, especially in rural and telemedicine settings, is a significant barrier to accurate diagnosis and automated systems.Inspired by the AIROGS challenge, this project offers a robust deep learning-based glaucoma screening system designed to address these problems. Using Generative Adversarial Networks (GANs) and super-resolution techniques, it enhances low-quality retinal images, restoring important details needed for reliable analysis.The model can function consistently across datasets with different image distributions and quality levels thanks to the integration of a gradient reversal layer for domain adaptation.The classification pipeline combines Vision Transformers (ViT) to capture intricate retinal structures and global dependencies with EfficientNet-B0 for effective feature extraction. The system showed excellent accuracy and adaptability during training and testing on the Zenodo dataset, indicating that it is ready for practical implementation. By reducing reliance on clinical expertise, this automated and scalable solution makes glaucoma detection more accessible and early in a variety of healthcare settings.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLE

CHAPTER 1

INTRODUCTION

The project is to develop a user-friendly software application that meets specific functional and system requirements while running efficiently on minimal hardware. Designed to be compatible with widely used operating systems like Windows 7, 8, 8.1, and 10, the software leverages Python as the primary programming language due to its simplicity, flexibility, and extensive library support. With modest hardware requirements, including a Pentium Dual Core processor, 2GB of RAM, and 120GB of hard disk space, the project ensures accessibility even on lower-end systems. This makes the application both practical and scalable, ideal for educational, research, or small-scale business purposes. The clear separation of hardware and software requirements also aids in easy deployment and maintenance.

1.1 OBJECTIVE OF THE PROJECT

The objective of this project is to create an efficient AI-based glaucoma screening system using color fundus images to detect glaucoma early and accurately. Efficient deep learning modules such as EfficientNet-B0 for efficient local feature extraction, Vision Transformers (ViT) for modeling global spatial relationships, and domain adaptation mechanics to generalize across a very wide range of imaging conditions are used in the proposed system to eliminate manual diagnosis. Super-resolution methods and Generative Adversarial Networks (GANs) are also used to improve the quality of noisy and low-resolution images. The main objective is to provide an automatic, accurate, and scalable diagnostic tool to work under a very wide range of real-world conditions, particularly in telemedicine and remote areas where specialist access is limited. The project addresses image variability and domain shifts to make eye care accessible and equitable.

1.2 SCOPE OF THE PROJECT

The Main work covers everything from preprocessed data to the deployment of AI models in developing a glaucoma screening system. It combines techniques for high classification accuracy, model generalization, and image quality improvement.

1.2.1 Data Collection and Processing

The key elements of this glaucoma screening project start with data collection and preprocessing, wherein the fundus images are downloaded from the publicly available Zenodo dataset. The images are subjected to a series of preprocessing operations, including resizing, grayscale normalization, and super-resolution image enhancement through the assistance of Generative Adversarial Networks (GANs). Besides ensuring equal quality input, this preprocessing has been done to pre-process data for proper model training.

1.2.2 Model Architecture

The model architecture is designed as a hybrid deep learning model that exploits multiple state-of-the-art elements. EfficientNet-B0 is utilized for efficient but lightweight local feature extraction, and Vision Transformers (ViT) are utilized to extract global spatial context from the fundus images. For further low-quality input improvement, visual enhancement is carried out using GANs. Domain adaptation is also utilized using a Gradient Reversal Layer (GRL) and a domain discriminator to facilitate the model to generalize across images from various sources and screening settings.

1.2.3 Training and Evaluation

The training and testing stage, the model is trained with supervised learning with labeled images annotated on them as referable glaucoma (RG) or non-referable glaucoma (NRG), respectively. Various optimization methods such as data augmentation and early stopping are used to prevent overfitting. The model is tested for performance using standard metrics such as accuracy, area under the curve (AUC), precision, and recall.

1.2.4 Deployment and Accessibility

The deployment and accessibility aspect of the project involves the application of the model learned in real-world conditions. The system can be deployed in hospitals as well as in telemedicine and rural areas. The robustness to variations in image quality implies that it can give proper results even under poor imaging conditions, thereby allowing early and mass screening of glaucoma.

1.3 IMPORTANCE OF THE PROJECT

The project is instrumental to the democratization of eye care, especially in areas where minimal access to trained ophthalmologists is available. Glaucoma, a leading cause of irreversible blindness, requires early detection. The slow traditional diagnosis is dependent on the expert. This bottleneck is avoided by the use of deep learning, allowing for scalable, automated analysis.The system is rendered robust to variations in image quality, a day-to-day reality in real-world datasets, by the combination of GANs and domain adaptation. Utilizing ViTs and EfficientNet-B0, hybrid model structure ensures local and global patterns are learned, enhancing the reliability of the diagnosis.Lastly, this work not only speeds up the screening but also lightens the workload for healthcare workers, encourages early interventions, and prevents preventable vision loss. Its use in telemedicine and mobile settings extends the reach of preventive healthcare and strengthens AI's position in contemporary ophthalmology.

1.4 ORGANIZATION OF THE PROJECT

To ensure effective guidance, timely implementation, and effective execution, the glaucoma screening project is systematically divided into phases. Each phase is planned with defined objectives, timelines, and accountability for maintaining systematic progress. Periodic review and stakeholder involvement are built in to ensure responsiveness and alignment during the project life cycle.

1.4.1 Planning & Resource Allocation

To guarantee clarity, methodical advancement, and successful execution, the glaucoma screening project is divided into clearly defined phases. Planning and resource allocation are the first steps, during which precise objectives, deadlines, and deliverables are set. To expedite execution, duties are divided among several important areas, such as data handling, model development, testing, and deployment.

1.4.2 Data Preparation

The Fundus images are gathered from the Zenodo dataset in the next phase, data preparation, which is followed by preprocessing techniques like noise reduction, resizing, normalization, and super-resolution to improve image quality and consistency.

1.4.3 Model Development

In The model development phase involves building a hybrid deep learning architecture that combines Vision Transformers to model global spatial dependencies, GANs for image enhancement, EfficientNet-B0 for effective local feature extraction, and domain adaptation elements like Gradient Reversal Layers (GRL) and domain discriminators to reduce domain-specific biases.

1.4.4 Model Training & Performance Evaluation

The In the model training and performance evaluation phase, a comprehensive model is subsequently trained on images classified as non-referable glaucoma (NRG) and referable glaucoma (RG) using supervised learning techniques. Standard performance metrics such as accuracy, precision, recall, and AUC are used for evaluation, with particular focus on the model's generalizability under a variety of difficult real-world circumstances.

1.4.5 Deployment and Outcome

Finally In order to enable real-time screening in clinical and remote settings, the trained model must be integrated into cloud-based or mobile platforms during the deployment and outcome phase. In addition to undergoing final validation to ensure that it is ready for clinical use, the model is updated frequently with new data to allow for ongoing learning. This methodical, phased approach guarantees that the project will affect early glaucoma detection and advance AI-driven solutions in preventive ophthalmology in addition to being efficient and scalable**.**

CHAPTER 2

LITERATURE SURVEY

2.1. A human-centered evaluation of a deep learning system deployed in clinics for the detection of diabetic retinopathy [1]

Author: Angelica CScanzera , CameronBeversluis, ArchitVPotharazu.

Year: 2023

This Diabetic retinopathy (DR) is a leading cause of vision loss in the United States and throughout the world. With early detection and treatment, sight-threatening sequelae from DR can be prevented. Although artificial intelligence (AI) based DR screening programs have been proven to be effective in identifying patients at high risk of vision loss, adoption of AI in clinical practice has been slow. We adapted the United Kingdom Design Council's Double-Diamond model to design a strategy for care delivery which integrates an AI-based screening program for DR into a primary care setting. Methods from human-centered design were used to develop a strategy for implementation informed by context-specific barriers and facilitators. The purpose of this community case study is to present findings from this work in progress, including a system of protocols, educational documents and workflows created using key stakeholder input.

2.2.1 Limitations

AI integration must adapt to diverse clinic settings, which can lead to inconsistent implementation and effectiveness.

2.2. UNETR: Transformers for 3D medical image segmentation [2]

Author: Prottoy Saha, Muhammad Sheikh Sadi, Md. Atikur Rahman.

Year: 2022.

Fully Convolutional Neural Networks (FCNNs) with contracting and expanding paths have shown prominence for the majority of medical image segmentation applications since the past decade. In FCNNs, the encoder plays an integral role by learning both global and local features and contextual representations which can be utilized for semantic output prediction by the decoder. Despite their success, the locality of convolutional layers in FCNNs, limits the capability of learning long-range spatial dependencies. Inspired by the recent success of transformers for Natural Language Processing (NLP) in long-range sequence learning, we reformulate the task of volumetric (3D) medical image segmentation as a sequence-to-sequence prediction problem. We introduce a novel architecture, dubbed as UNEt TRansformers (UNETR), that utilizes a transformer as the encoder to learn sequence representations of the input volume and effectively capture the global multi-scale information, while also following the successful" U-shaped" network design for the encoder and decoder.

2.2.2Limitations

Models trained on specific datasets may struggle with generalization when applied to different imaging modalities or patient populations.

2.3. Self-supervised pre-training of Swin transformers for 3D medical image analysis [3]

Author: Tang, Yucheng, Dong Yang, Wenqi Li, Holger R. Roth.

Year: 2022.

Vision Transformers (ViT) s have shown great performance in self-supervised learning of global and local representations that can be transferred to downstream applications. Inspired by these results, we introduce a novel self-supervised learning framework with tailored proxy tasks for medical image analysis. Specifically, we propose:(i) a new 3D transformer-based model, dubbed Swin UNEt TRansformers (Swin UNETR), with a hierarchical encoder for self-supervised pre-training;(ii) tailored proxy tasks for learning the underlying pattern of human anatomy. We demonstrate successful pre-training of the proposed model on 5050 publicly available computed tomography (CT) images from various body organs. The effectiveness of our approach is validated by fine-tuning the pre-trained models on the Beyond the Cranial Vault (BTCV) Segmentation Challenge with 13 abdominal organs and segmentation tasks from the Medical Segmentation Decathlon (MSD) dataset. Our model is currently the state-of-the-art on the public test leaderboards of both MSD and BTCV datasets.

2.2.3 Limitations

Training Swin UNETR requires significant GPU resources, making it less accessible for researchers with limited hardware.

2.4. Medical image segmentation using deep learning: A survey [4]

Author: Martins E. Irhebhude, Oladimeji A. Adeyemi.

Year: 2022.

It categorizes segmentation methods into fully supervised, weakly supervised, and unsupervised approaches, emphasizing the role of CNNs, U-Net, DeepLab, and transformer-based architectures in enhancing segmentation accuracy. The paper highlights the importance of network design**,** including skip connections, attention mechanisms, and multi-scale feature extraction, to improve performance. Additionally, it explores loss function improvements, such **as** Dice loss, focal loss, and boundary-aware losses, which help address class imbalances common in medical images. The survey examines data augmentation and transfer learning as strategies to enhance robustness, especially when dealing with limited labeled datasets. A key focus is the challenge of, ensuring deep learning models generalize across different domain adaptation imaging modalities and clinical environments. The authors also discuss the interpretability concerns surrounding AI-based segmentation, emphasizing the need for explainable AI techniques in healthcare applications.

2.2.4 Limitations

Models trained on specific datasets may struggle with generalization when applied to different imaging modalities or patient populations.

2.5. A workflow for computer-aided diagnosis of glaucoma [5]

Author: Wang, H., Sun, H., Fang, Y., Li, S., Feng, M., & Wang.

Year: 2022.

Glaucoma is an optic disease with visual impairment, leading to blindness. The timely diagnosis of glaucoma can reserve the blinding process with a good prognosis. Although typical glaucoma is characterized by elevated intraocular pressure and optic neuropathy, accurate diagnosis is still challenging by measuring intraocular pressure and ophthalmoscopy diagnosis. This paper presents a workflow to diagnose glaucoma and distinguish low-quality fundus images automatically. The performance in the hidden test set shows the promising clinical usefulness against experts, which achieved first place in the AIROGS challenge with the partial AUC (90%~100% specificity) of 0.8998 and sensitivity of 0.8533 at 95% specificity.

2.2.5 Limitations

Deep learning models function as black boxes, making it difficult for clinicians to understand AI-generated predictions.

2.6. Deep Dirichlet uncertainty for unsupervised out-of-distribution detection of eye fundus photographs in glaucoma screening [6]

Author: Araújo, T., Aresta, G., & Bogunović, H.

Year: 2022.

The development of automatic tools for early glaucoma diagnosis with color fundus photographs can significantly re-duce the impact of this disease. However, current state-of-the-art solutions are not robust to real-world scenarios, providing over-confident predictions for out-of-distribution cases. With this in mind, we propose a model based on the Dirich-let distribution that allows to obtain class-wise probabilities together with an uncertainty estimation without exposure to out-of-distribution cases. We demonstrate our approach on the AIROGS challenge, where we achieve a performance sim-ilar to other participants without requiring additional annotations or artificially generated out-of-distribution labels.

2.2.6 Limitations

While effective in detecting out-of-distribution (OOD) cases, the model may struggle with unseen imaging modalities or variations in fundus photographs.

2.7. Training data-efficient image transformers & distillation through attention [7]

Author: Touvron, Hugo, Matthieu Cord, Matthijs Douze,

Year: 2021.

Recently, neural networks purely based on attention were shown to address image understanding tasks such as image classification. These high-performing vision transformers are pre-trained with hundreds of millions of images using a large infrastructure, thereby limiting their adoption. In this work, we produce competitive convolution-free transformers trained on ImageNet only using a single computer in less than 3 days. Our reference vision transformer (86M parameters) achieves top-1 accuracy of 83.1%(single-crop) on ImageNet with no external data. We also introduce a teacher-student strategy specific to transformers. It relies on a distillation token ensuring that the student learns from the teacher through attention, typically from a convnet teacher. The learned transformers are competitive (85.2% top-1 acc.) with the state of the art on ImageNet, and similarly when transferred to other tasks. We will share our code and models.

2.2.7 Limitations

Despite improvements, transformers still require substantial data for optimal performance, making them less effective for low-data regimes.

2.8. Deep learning and glaucoma specialists: The relative importance of optic disc features to predict glaucoma referral in fundus photographs [8]

Author: Phene, S., Dunn, R. C., Hammel, N., Liu, Y., Krause.

Year: 2019

This paper presents a segment-annotation-free method for the recognition of handwritten Chinese text, addressing the limitations of traditional segmentation-based approaches that require manually annotated segment boundaries. Chinese handwritten text is especially difficult to segment because of its high character density, wide range of writing styles, and irregular word boundaries. The authors suggest a deep learning-based framework that uses weak supervision to learn efficient segmentation and recognition without the need for explicit segment annotations in order to get around these issues. The system makes use of a multi-stage pipeline that combines segmentation-prediction module that uses a segment-free loss function to align predictions with ground truth text with convolutional neural networks (CNNs) for feature extraction. The alignment-free learning mechanism, which eliminates the need to specify individual character positions and allows the model to be trained using only sequence-level annotations (i.e., the entire transcribed text), is the main innovation.

2.2.8 Limitations

The dataset was primarily from Chinese populations, raising concerns about the model's performance across diverse ethnicities and imaging conditions

2.9. Prevalence of end-of-life visual impairment in patients followed for glaucoma [9]

Author: Michiel J W M Busch, Carroll A B Webers, Henny

Year: 2018

This study conducted in the Netherlands assessed the prevalence of end-of-life visual impairment in patients followed for glaucoma2. The research analyzed data from 122 patients who had passed away between July 2008 and July 2010, all of whom had been monitored for glaucoma. The findings revealed that among patients with open-angle glaucoma (OAG)**,** 26% experienced visual impairment at the end of life**,** with 15% of cases directly attributed to glaucoma2. Notably, substantial visual loss at the initial visit was a significant contributing factor. In contrast, among glaucoma suspects or patients with ocular hypertension (OHT), the prevalence of visual impairment was lower (8%).

2.2.9 Limitations

Studies on end-of-life visual impairment often rely on retrospective data, which may lack comprehensive follow-ups or standardized records, affecting accuracy.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The Conventional optical character recognition (OCR) engines or segmentation-based deep learning techniques are used by the majority of handwritten document recognition systems today. These models typically require documents to be preprocessed into individual lines or words using segmentation algorithms such as Projection Profiling or Connected Component Analysis (CCA) before they can be recognized. Although such methods have achieved a respectable level of accuracy on clean, modern printed text, they suffer greatly when applied to historical manuscripts, which often have irregular layouts, faded ink, varying handwriting styles, and deteriorated document quality. Additionally, certain existing deep learning models, such as Recurrent Neural Networks (RNNs) and CNN-LSTM hybrids, are limited by their sequential processing architecture and dependence on manually segmented data. These systems frequently interpret documents incorrectly and are susceptible to segmentation errors with non-linear reading orders, multi-column formats, or annotations. Additionally, because they rely on character-level annotations during training, which increases the need for laborious data preparation, they are less scalable for large historical archives. Because they often lead to incomplete transcriptions, misinterpret structural elements, or require extensive manual correction, these limitations diminish the effectiveness of current solutions in digitization and preservation workflows. Segmentation-free, layout-aware recognition models, such as the proposed Document Attention Network

(DAN), are crucial for processing full-page handwritten documents accurately and comprehensively without the need for manual intervention.

3.2 PROPOSED SYSTEM

The Our suggested system is a clever, AI-powered way to use color fundus photos of the eye to identify glaucoma early. It is designed to function consistently under real-world circumstances, even when the input images are of varying quality because of various cameras, lighting, or surroundings. The proposed system is inspired by the work of F. Khader, titled *"Elevating Fundoscopic Evaluation to Expert Level – Automatic Glaucoma Detection Using Data from the AIROGS Challenge"*. This system aims to automate the process of glaucoma detection using deep learning techniques applied to retinal fundus photographs. Leveraging high-quality, annotated data from the AIROGS (Artificial Intelligence for RObust Glaucoma Screening) challenge, the model is trained to identify glaucomatous features at a level comparable to expert ophthalmologists[10]. We employ methods like a Gradient Reversal Layer (GRL) and a domain discriminator because images can vary greatly depending on the source (for instance, different hospitals may use different cameras). These ensure consistent accuracy across all inputs by assisting the model in ignoring image-source variations and concentrating only on the important medical features.We combine the capabilities of two potent models for image analysis: Vision Transformers (ViT) for comprehending the image's larger structure and relationships, and EfficientNet-B0 for lightweight and quick feature extraction. When combined, they enable the system to produce extremely precise forecasts.Following analysis, the system assigns the image to one of two groups: Referable Non-Referable Glaucoma (NRG), which means no significant signs are found, or Glaucoma (RG), which indicates signs that require medical attention. This outcome can then be shown to medical professionals or even incorporated into telemedicine platforms. All things considered, the suggested system is designed to be accurate, scalable, and useful, which makes it an invaluable instrument for early glaucoma screening, particularly in places with limited access to eye care professionals.

3.3 FEASIBILITY STUDY

Before beginning to develop this glaucoma screening system, we took a step back to determine whether it was technically and practically feasible. A feasibility study can help with that. It assisted us in determining whether the concept could be effectively developed into a practical solution that offers genuine value, particularly in actual medical settings. Technically speaking, the project is very doable. We have all we need to construct and operate the system efficiently thanks to the availability of sophisticated deep learning frameworks, openly available datasets like Zenodo, and effective hardware like the Arduino Nano 33 BLE or other embedded systems. Additionally, model architectures like EfficientNet and Vision Transformers, as well as cloud platforms like Edge Impulse, make model deployment and training easier easier to handle, even with fewer resources. This project is reasonably cost-effective in terms of economic viability. Our use of free datasets, open-source tools, and inexpensive hardware components makes it feasible for both scalable deployment and research. After training, the model can operate on portable devices, which makes it perfect for mobile screening units or rural health centers with limited funding.Operationally, the system works well with the current workflows in healthcare. By automating the initial screening process, it supports physicians rather than replaces them. This can cut down on workload, save time, and identify possible cases sooner. Even with little training, healthcare professionals can easily use it because the results are straightforward to interpret (glaucoma vs. non-glaucoma). Finally, with regard to Given its social and clinical significance, early glaucoma detection is imperative. Simply because they are unaware that they have the condition, many people lose their vision. This project could significantly improve public health by increasing screening's efficiency and accessibility, particularly in underserved areas. In conclusion, the project is not only feasible but also well-positioned to provide genuine benefits to patients and healthcare providers, according to our assessment of technical, financial, and practical factors.

3.3.1 Economical feasibility

The project's cost-effectiveness and accessibility make it a good fit for a range of healthcare settings, particularly in under-resourced and rural areas.The hardware needs are low and reasonably priced. In order to run the AI model, the system primarily uses a standard fundus camera, which is already available in many eye clinics. It can also be integrated with low-cost computing platforms like the Raspberry Pi or Arduino Nano 33 BLE. When compared to conventional diagnostic tools, these parts are easily accessible and reasonably priced.We use open-source software tools like Python, TensorFlow, and Edge Impulse, which do not require expensive software licenses. Once trained, the deep learning models can be used again without constant costs, and new data can be added as needed over time.Additionally, the system lessens the workload for ophthalmologists by automating the screening process, which eliminates the need for manual patient diagnosis. Long-term cost savings are substantial as a result, particularly in situations involving high screening volumes. All things considered, the project provides a high return on investment because it lowers development and deployment costs while improving early glaucoma detection. This makes it a financially viable option for small clinics, telemedicine projects, and public health programs alike.

3.3.2 Technical feaisibility

The proposed system for automated glaucoma detection is technically feasible through the integration of advanced deep learning models, specifically Vision Transformers (ViT), as introduced by A. Dosovitskiy in *"An Image is Worth 16×16 Words"*. The Vision Transformer architecture represents a significant shift from traditional convolutional neural networks (CNNs) by applying the transformer mechanism—originally developed for natural language processing—directly to image data[11]. Standard computer resources, such as laptops with GPU support or cloud services like Google Colab, can be used to train and test these models.

Additionally popular and supported by existing libraries are Generative Adversarial Networks (GANs) and super-resolution algorithms for image enhancement. These methods can function effectively on contemporary hardware without the need for costly or specialized equipment.Additionally, we have employed domain adaptation strategies, such as the Gradient Reversal Layer (GRL), which are both technically sound and simple to incorporate into conventional model architectures. These support our system's resilience across a variety of datasets, which is essential for practical application. Our model is trained and evaluated on the Zenodo Glaucoma Dataset, which is freely accessible and full of varied examples. The feasibility is further enhanced by the fact that no proprietary data or expensive clinical imaging equipment is required.

3.3.3 Operational feasibility

From From an operational perspective, the suggested glaucoma screening system is very doable and useful to use in actual environments. From taking pictures to detecting glaucoma, the entire procedure is made to run smoothly with little manual involvement. The system automatically processes the fundus image after it is taken, improves its quality if necessary, and outputs a diagnostic result. Because of this, the solution can be used even in settings with a shortage of medical personnel or knowledge.The system can handle inputs from various clinics, cameras, or locations because it is based on deep learning and was trained using a variety of retinal images. This makes it suitable for both urban hospitals and rural health centers. Furthermore, the system can operate effectively on standard computing power thanks to the use of lightweight models like EfficientNet-B0 gadgets that do not require costly hardware.The model's ability to be seamlessly incorporated into current workflows, like telemedicine platforms or mobile screening units, is another significant benefit. A straightforward user interface makes it easy to interpret results and train health professionals to use the system.The system is generally dependable, simple to use, and requires little continuous maintenance, which makes it operationally viable for widespread implementation in both private healthcare settings and public health campaigns.

3.3.4 Social feasibility

This Because it tackles a pressing public health issue—the early detection of glaucoma, one of the main causes of irreversible blindness globally—the project has strong social viability. Many people lack easy access to trained ophthalmologists and routine eye exams, particularly in underserved or rural areas. This project fills that gap and makes necessary eye care more accessible by offering an automated, AI-powered screening tool. The system is simple to use, non-invasive, and doesn't require any sophisticated technical expertise to function. Health professionals can use it to screen patients and determine who might require additional medical care, even with only rudimentary training. By encouraging early diagnosis and treatment, this strengthens communities and can drastically lower the danger of irreversible vision loss. By making cutting-edge technology available outside of large cities and hospitals, the project fosters social inclusion and healthcare equity. Additionally. Over time, this may result in a higher standard of living and less financial strain on families and healthcare systems.

Overall, the project is very feasible and significant from a societal standpoint because it is people-friendly, socially conscious, and in line with public health objectives.

3.4 LANGUAGE SPECIFICATION – PYTHON

With good reason, we decided to use Python as the main programming language for this project. Python is well-known for its ease of use, readability, and robust libraries, which make it an ideal choice for image processing and machine learning applications such as glaucoma detection.We developed, trained, and assessed our glaucoma screening model using Python's strong deep learning support, which includes well-known libraries like TensorFlow, Keras, and PyTorch. These libraries provide pre-built components that give us greater flexibility and significantly less code when designing complex neural networks, such as EfficientNet and Vision Transformers.Additionally, we prepared the dataset, resized images, and normalized pixel values using Python's image handling and preprocessing tools, including OpenCV, PIL (Pillow), and NumPy. Libraries such as these are useful for training performance and visualizing results Seaborn and Matplotlib were very helpful.Beyond machine learning, Python's clear syntax and robust community made it simple to plan the entire project workflow, from data loading and training to prediction and evaluation.To put it briefly, Python not only aided in the creation of an intelligent and trustworthy glaucoma detection system, but it also sped up, eased, and improved the development process.Additionally, we prepared the dataset, resized images, and normalized pixel values using Python's image handling and preprocessing tools, including OpenCV, PIL (Pillow), and NumPy. Libraries like Matplotlib and Seaborn were very helpful for training performance and visualizing results.Beyond machine learning, Python's clear syntax and extensive feature set made it simple to plan the entire project workflow, from data loading and training to prediction.

3.4.1 Advantages of using python

Python proved to be a great option for our glaucoma screening project because of its many benefits. Python's strong machine learning capabilities are among its greatest advantages. Tools like PyTorch, TensorFlow, and Keras have made it simple to create and train sophisticated deep learning models like Vision Transformers and EfficientNet-B0. It also performs exceptionally well in image processing, where libraries such as OpenCV, PIL, and NumPy enabled us to effectively manage fundus image preprocessing tasks like resizing, normalization, and enhancement—all of which are essential steps in enhancing image quality. Furthermore, our development was significantly accelerated by Python's sizable developer community and abundance of online resources. We were able to quickly locate tutorials and solutions for any problems we ran into, which increased productivity and saved time. Because of its cross-platform compatibility, our code was able to It is convenient for both local development and deployment on servers or embedded devices because it runs smoothly on Windows, macOS, and Linux.

3.4.2 Reliability

One Our glaucoma screening system's dependability—its capacity to function reliably and precisely under various circumstances—is one of its main advantages. We built the system from the ground up to tackle real-world issues like low image quality, erratic lighting, and photos captured by various fundus camera types.The system can correctly identify glaucoma symptoms in images it has never seen before by utilizing domain adaptation techniques in conjunction with sophisticated deep learning models like EfficientNet-B0 and Vision Transformers. Because of this, it can be relied upon in clinical and field settings, including remote locations with scarce medical resources. Our development language, Python, also helps to ensure this dependability. Its robust error-handling, robust libraries, and sizable developer community all contribute to the system's seamless operation and is simple to maintain and debug. Furthermore, we validated our model's performance with real-world data by training and testing it on a well-labeled dataset from Zenodo.

Overall, the system is a reliable tool for early glaucoma detection and blindness prevention because it is designed to perform consistently and reliably in real-world healthcare settings in addition to having high accuracy under ideal circumstances.

CHAPTER 4

SYSTEMDESIGN

4.1 SYSTEM ARCHITECTURE

The In both clinical and telemedicine settings, the glaucoma screening system is built as an effective and scalable pipeline made up of several interconnected modules, each of which carries out a specific task to guarantee precise and trustworthy diagnosis. The first step in the process is the Input Layer, where fundus images are obtained from public datasets such as Zenodo or screening devices. The size and quality of these photos frequently vary. Next, By resizing the images to a fixed size and using normalization techniques to adjust contrast and brightness, the Preprocessing Module standardizes the images.

The system uses super-resolution techniques and GAN-based enhancement to improve the clarity of lower-quality images.To make sure the model learns source-invariant features, the Domain Adaptation Layer employs a Gradient Reversal Layer (GRL) and a domain discriminator after preprocessing. This stage improves the model's ability to generalize across various clinical settings and devices.

The A hybrid deep learning architecture is then used by the Feature Extraction Layer to process the previously processed images. While

Vision Transformers (ViT) capture the image's global spatial context and relationships, EfficientNet-B0 is used to capture fine-grained, high-resolution features locally.The The Classification Layer receives the extracted features and uses a fully connected neural network to predict either Non-Referable Glaucoma (NRG) or Referable Glaucoma (RG).

Finally, In addition to displaying the classification result, the output layer has the ability to integrate with a medical records database or initiate an alert system.The system is flexible in terms of deployment; it can be incorporated into telemedicine applications or set up on cloud platforms for extensive screenings. Additionally, it can be used offline, which makes it appropriate for use in environments with limited resources or in rural areas.

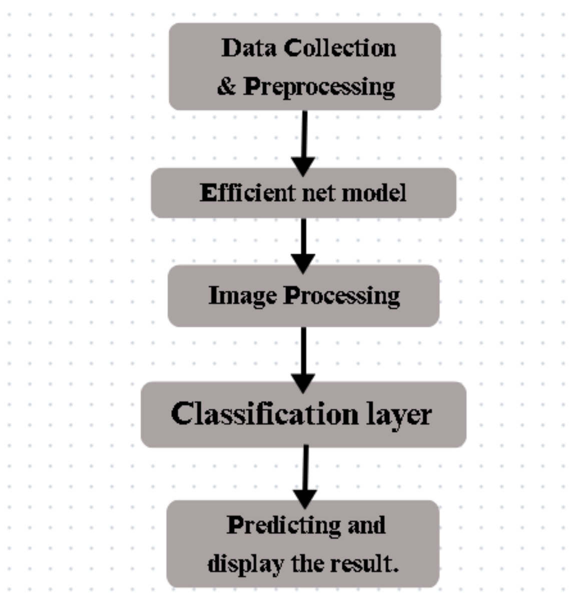


Figure 4.1 Work Flow Architecture

The detailed procedure used in our glaucoma detection system is shown in **Figure 4.1** To guarantee consistent quality, fundus images are collected, resized, normalized. The EfficientNet model, a small yet effective deep learning architecture, including the optic disc and cup. Following feature extraction, image processing is applied to the data, where the features are further honed and organized for classification. The classification layer receives the processed data and decides whether or not glaucoma is present in the input image. The system then advances to the prediction and display phase, where the outcome—whether referable or non-referable glaucoma—is displayed to the user, completing a fast and automated screening process.

Table 4.1: Core Components of the Glaucoma Detection Architecture

Component	**Description**	**Technology Used**
Preprocessing	Resizes images, applies normalization, and enhances image clarity	OpenCV, PIL, GAN, Super-Resolution
Feature Extractor 1	Extracts local features from the fundus image	EfficientNet-B0
Feature Extractor 2	Captures global spatial relationships within the image	T5 Transformer
Feature Fusion	Combines local and global features into a unified representation	PyTorch, Flask

The According to **Table 4.1**, the procedure starts with a preprocessing step in which each fundus image is resized, normalized, and improved using programs like PIL and OpenCV as well as deep learning-based methods like GANs and super-resolution. This step guarantees that the model can use even lower-quality images efficiently. After that, the preprocessed photos are run through two different feature extractors. The first is EfficientNet-B0, which concentrates on obtaining local, fine-grained features from the image, like the details surrounding the cup and optic disc. The second is a Vision Transformer, which helps the system comprehend the overall picture of retinal health by capturing global spatial relationships throughout the entire image.A rich, cohesive representation of the input is produced by fusing the two sets of features together using a feature aggregation technique after they have been extracted. The classification head, a fully connected neural layer constructed with PyTorch, receives this combined data and decides whether the image exhibits symptoms of Non-Referable Glaucoma (NRG) or Referable Glaucoma (RG). Lastly, the model is linked to a web interface built on Flask, which enables users to upload images with ease and get real-time predictions. Because the entire system is built for real-world implementation, it is dependable, quick, and accessible in clinical and remote healthcare settings.

CHAPTER 5
MODULE DESCRIPTION
5.1 DATA COLLECTION

In In this project, a deep learning model created with Python forms the basis of our glaucoma screening system. Python was selected because it offers robust, user-friendly libraries ideal for image processing, machine learning, and creating AI-powered medical systems.

Our The model uses a hybrid strategy for feature extraction, combining Vision Transformers (ViT) and EfficientNet-B0. This combination enables us to record both global image patterns (such as the structural relationships in the eye) and local retinal features (such as the optic cup and disc), both of which are crucial for identifying glaucoma symptoms.

5.1.1 Model Architecture

The A hybrid deep learning architecture is used by the glaucoma screening system to extract and classify features. Fine-grained, high-resolution features are extracted from fundus images using EfficientNet-B0. Because of its reputation for being computationally efficient and lightweight, it is perfect for applications with constrained processing or data resources. The global structure of the eye and spatial dependencies are simultaneously captured by Vision Transformers (ViTs), which offer a more comprehensive contextual understanding to supplement the localized features obtained by EfficientNet-B0.

The A thorough feature representation is then created by concatenating the outputs from the Vision Transformer and EfficientNet-B0. A fully connected layer receives this combined feature vector and uses it to perform the final classification, determining whether the input image shows Non-Referable Glaucoma (NRG) or Referable Glaucoma (RG). By combining local detail with global context, this dual-stream approach improves the model's accuracy and robustness.

5.1.2 Data handling

In Python libraries like OpenCV, NumPy, and Pillow (PIL) are used to load and process fundus images for the glaucoma screening system. In order to ensure that the images are in the best possible format for model input, these libraries are necessary for tasks like image reading, resizing, normalization, and enhancement.

To The dataset is divided into training, validation, and testing sets in order to efficiently train and assess the model. A balanced and methodical data division that supports model training, hyperparameter tuning, and performance evaluation on unseen data is made possible by tools like scikit-learn's train_test_split function or custom Python scripts.

5.1.3 Training process

The Python is the main scripting language used to manage the training pipeline for the glaucoma screening model, which is trained using TensorFlow or PyTorch. During training, data augmentation methods like flipping, rotation, and brightness adjustment are used to improve the model's capacity to generalize to new data. These additions make the model more resilient to changes in lighting and image orientation.

5.1.4 Evaluation

The Scikit-learn's Python metric functions are used to assess the glaucoma screening model's performance. Accuracy, precision, recall, and AUC (area under the curve) are important evaluation metrics. These metrics offer a thorough understanding of the classification efficacy of the model, particularly with regard to differentiating between cases of Referable Glaucoma (RG) and Non-Referable Glaucoma (NRG). Visualization tools such as Seaborn and Matplotlib are used to convey results and obtain additional insights. These libraries facilitate the creation of educational plots that make it simpler to understand the behavior of the model and identify areas for development, such as confusion matrices, ROC curves, and performance trend graphs.

5.1.5 Real-world Integration

In Python scripts are used in the implementation of the glaucoma screening system to create a comprehensive end-to-end pipeline that can take an input fundus image and generate a classification result in real time. This makes the system appropriate for situations involving remote diagnosis as well as clinical use.

For Lightweight web frameworks like Flask or FastAPI can be used to integrate the system with front-end interfaces or telemedicine platforms for smooth user interaction and deployment, which improve accessibility and usability in real-world scenarios by enabling the model to instantly return predictions after receiving image inputs from web or mobile applications.

5.2 FEATURE EXTRACTION AND SELECTION

In Important steps in our glaucoma screening system are feature extraction and selection, which aid the model in identifying the key elements of fundus images associated with glaucoma. Python and robust deep learning libraries like TensorFlow, Keras, and PyTorch were used to effectively implement these steps.

5.2.1 Feature Extraction

For We used a hybrid deep learning strategy that combined Vision Transformers (ViT) and EfficientNet-B0, a lightweight convolutional neural network, for this glaucoma screening task. Prior to being optimized for our particular glaucoma dataset, both models were pre-trained on extensive image datasets to take advantage of pre-existing feature representations. Local features like edges, blood vessels, and the structural details of the optic disc—all crucial components in the detection of glaucoma—were captured by EfficientNet-B0. Vision Transformers, on the other hand, concentrated on learning the relationships between various regions and comprehending the image's global context. This is especially helpful for spotting dispersed and subtle glaucoma symptoms that might not be visible from local characteristics alone. This combination improved the system's capacity to classify data in a reliable and accurate manner.

5.2.2 Feature Selection

Once The fundus images are used to extract features, and feature selection is essential for focusing the model's attention on the most significant patterns, improving accuracy and efficiency. Python uses a number of methods to accomplish this.

Dropout layers are used in training to avoid overfitting, which encourages the model to overlook noisy or irrelevant features. In order to simplify the data without losing crucial insights, global average pooling is used to shrink feature maps while keeping the most significant information. When working with large or complex datasets, dimensionality reduction techniques such as Principal Component Analysis (PCA) are sometimes used to compress the feature space into its most informative components.

These The fully connected classification layer then receives the refined and chosen features and makes the ultimate determination—whether or not glaucoma is present in the input image. The model's accuracy and generalizability are maintained by this feature selection pipeline.

5.2.3 Python Tools Used

In Several Python libraries are crucial to data management, model integration, and analysis in the glaucoma screening pipeline. Smooth numerical computations and data manipulation are made possible by the effective handling of feature arrays and image data by NumPy and Pandas. TensorFlow, Keras, and PyTorch are examples of deep learning frameworks that make it easier to integrate models and give access to intermediate feature layers for additional processing. Scikit-learn provides helpful algorithms for feature selection, including Principal Component Analysis (PCA) and SelectKBest, which assist in locating and preserving the most informative features. Various plots and heatmaps are created using Matplotlib and Seaborn to visualize extracted features and model attention in order to better comprehend and interpret the model's behavior. This gives important information about how the system makes decisions.

5.3 DATA NORMALIZATION AND STANDARDIZATION

Before In order for our deep learning model to learn efficiently, we must prepare the data before feeding it the fundus images. Normalization and standardization are two crucial steps in this preparation. This increases the accuracy and stability of the model by ensuring that all input images have a similar scale.

5.3.1 Data Normalization

In Normalization is essential to our project because it modifies the fundus image pixel values. These pixel values typically fall between

0 and 255, but it's crucial to scale them down to a smaller range—typically between 0 and 1—for deep learning models.which could impair the model's ability to learn. Python frameworks like TensorFlow, Keras, or PyTorch can handle normalization automatically during data loading, or it can be done simply with libraries like NumPy. For instance, scaling pixel values to the [0, 1] range with NumPy only requires dividing the image array by 255.0.

5.3.2 Data Standardization

Ensure consistent and efficient training of deep learning models for glaucoma detection, the process of data standardization is critical. Inspired by the architecture proposed by G. Huang et al. in *"Densely Connected Convolutional Networks (DenseNet)"*, this system adopts standardized preprocessing techniques that complement the characteristics of densely connected networks.[12], NumPy makes standardization in Python simple. Standardized_image = (image - np.mean(image)) / np.std(image), for instance. As an alternative, you can use built-in transforms in well-known deep learning libraries like TensorFlow or PyTorch to apply standardization while loading data. For example, you can use a transform in PyTorch.Create a pipeline using transforms.To standardize, provide the mean and standard deviation values (which are frequently dataset-specific).

5.4 MODEL IMPROVISATION USING PYTHON

We created a simple deep learning model in the early phases of our project to divide fundus photos into two groups: glaucomatous and non-glaucomatous. Despite the model's encouraging results, it became evident that we needed to improve both its accuracy and generalization for real-world screening applications, especially when dealing with diverse or low-quality images from different sources and devices. Thankfully, Python's adaptability and vast library and tool ecosystem enabled us to refine our model iteratively. We were able to gradually improve the model's performance, robustness, and dependability for practical deployment in clinical and telemedicine settings by utilizing cutting-edge strategies like data augmentation, domain adaptation, model fine-tuning, and hybrid architectures.

5.4.1 Data Augmentation

Increase the robustness of our model, we used a variety of image augmentation techniques and robust libraries. These changes included cropping, zooming, brightness and contrast adjustments, rotation, and flipping both horizontally and vertically. We were able to improve the model's ability to identify glaucoma-related characteristics by exposing it to these variations during training, even in the face of minor variations in image orientation, lighting, or framing. This greatly improved the model's capacity to generalize across a variety of fundus images from the real world.

5.4.2 Replacing Base Model with EfficientNet-B0

In order to improve model performance, we switched from a conventional Convolutional Neural Network (CNN) to EfficientNet-B0, a cutting-edge architecture known for its accuracy and efficiency. This switch only needed a few lines of Python code thanks to TensorFlow/Keras, but it had a big impact. Better feature extraction from EfficientNet-B0 allowed the model to identify more pertinent patterns in fundus photos. Because of its optimized architecture, it also resulted in faster training times and a lower chance of overfitting, which is crucial when working with relatively small datasets. This improvement was a significant step toward strengthening and expanding our glaucoma screening model.

5.4.3 Introducing Vision Transformers (ViT)

We incorporated Hugging Face's Transformers library and Vision Transformers (ViT) into our pipeline using Python to improve the model's capacity to capture spatial relationships throughout the fundus image. Vision Transformers enabled the model to comprehend the global context and structural layout of the optic nerve region, which is crucial for an accurate diagnosis of glaucoma. This is in contrast to traditional CNNs, which concentrate on local features. This integration led to more dependable classification results and greatly enhanced the model's ability to identify subtle, dispersed patterns.

5.4.4 Domain Adaptation Improvement

In real-world medical imaging applications, domain shift is a common challenge due to variations in fundus cameras, lighting conditions, patient demographics, and clinical settings. To address this, the proposed system incorporates domain adaptation strategies inspired by the work of A. Galdran et al. in *“Open-Set Glaucoma*

Screening From Eye Fundus Images: Domain Knowledge to the Rescue" [13]. The GRL made it possible for the model to concentrate only on the clinical characteristics pertinent to glaucoma, ignoring domain-specific variations like variations in image resolution or lighting. This method aids in the model's improved generalization across a variety of sources and is a crucial component of domain adaptation. Because of Python's adaptability, we were able to easily incorporate this sophisticated method into our training pipeline by modifying the model's forward and backward passes to implement GRL.

5.4.5 Hyperparameter Tuning

Improve the performance of our model, we performed extensive hyperparameter tuning using optimization tools such as Optuna and Keras Tuner. We experimented with batch sizes, dropout rates, learning rates, and various optimizers like Adam and RMSprop. These seemingly insignificant changes, which were made entirely in Python, were essential in raising the model's validation accuracy and lowering overfitting. Through a methodical tuning procedure, we were able to determine the best setup for dependable and strong glaucoma detection.

CHAPTER 6

IMPLEMENTATION AND RESULT

The Torch installation procedure is shown in **Figure 6.1(a)**. We built, trained, and assessed the model for our glaucoma detection project using PyTorch, the central deep learning framework.

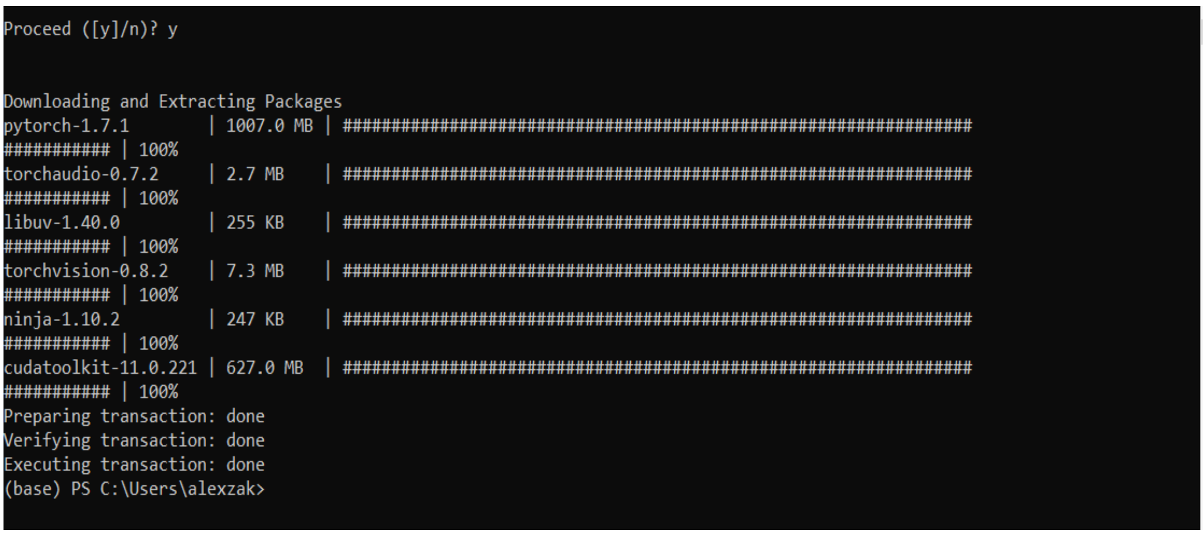


Figure 6.1(a) Installation of Pytorch

Because of its versatility and user-friendliness, PyTorch is a great option for research-based projects like ours. PyTorch's dynamic computation graph made it possible for us to freely test various model architectures, including EfficientNet-B0 and Vision Transformers, without having to redo intricate code.We were able to effectively prepare the fundus images for training by using PyTorch's robust tools for data augmentation, image preprocessing, and batch loading via torchvision. We adjusted our model during the training phase using PyTorch's optimizers and loss functions, and we continuously tracked performance metrics like accuracy, precision, and AUC.

Overall, PyTorch was crucial to the project's success because it allowed us complete control over every stage of the model development process, from loading data to deploying predictions.

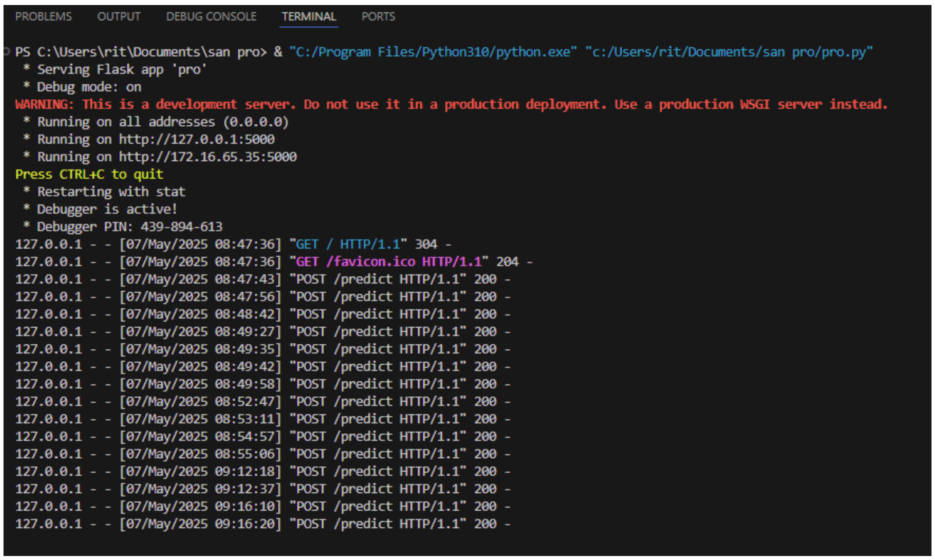


Figure 6.2 Model Training Progress

The **Figure 6.2** shows PyTorch was used to train the model for our glaucoma detection system, and a Flask web application was used to deploy the trained model. In order to increase variety, we first preprocessed fundus images by resizing them, normalizing pixel values, and using data augmentation. Referable Glaucoma (RG) and Non-Referable Glaucoma (NRG) were the labels assigned to the pictures. With these inputs, we trained a deep learning model (combining Vision Transformers and EfficientNet-B0) over a number of epochs, optimizing hyperparameters such as batch size and learning rate. From these labeled images, the model was able to identify characteristics associated with glaucoma, such as the optic cup and disc. After training, we saved the model and created an image upload interface using Flask. Behind the scenes, the Flask application loads the learned model, gets picture information through POST requests, and instantly provides predictions. Multiple successful predictions, as displayed in the terminal, show that the model is reacting appropriately, demonstrating the stability and efficacy of the training and deployment procedure.

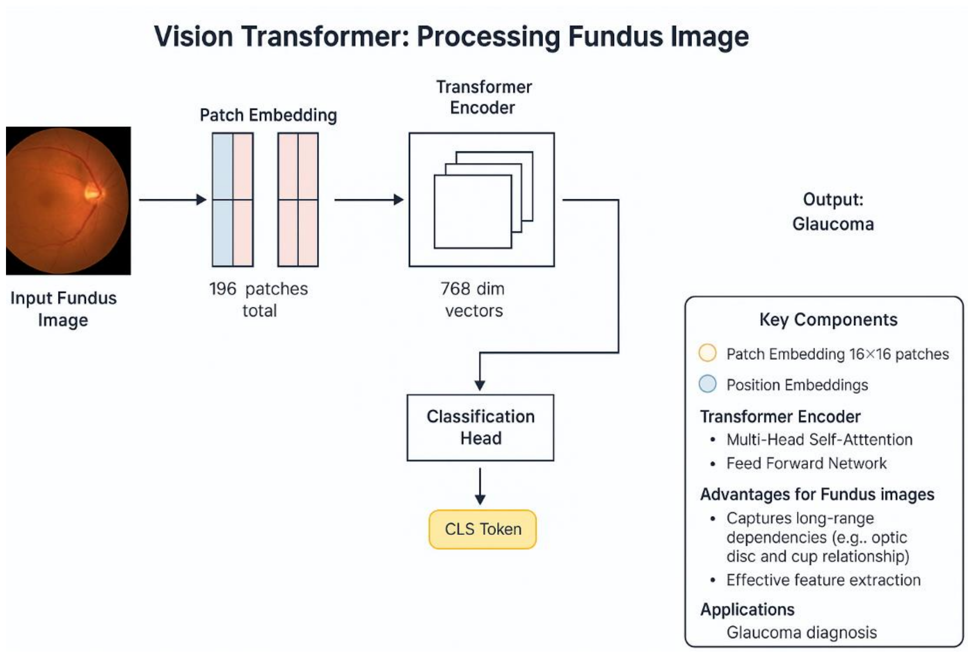


Figure 6.3 Vision Transformer Processing Pipeline

The processing of a fundus image by a Vision Transformer (ViT) to identify glaucoma symptoms is shown in **Figure 6.3**. An input fundus image is used as the starting point, and it is split up into 196 tiny patches, each measuring 16 by 16 pixels. After that, a patch embedding layer transforms these patches into numerical representations. Each patch embedding is supplemented with position embeddings to aid the model in comprehending the spatial arrangement of the image. The Transformer Encoder, which consists of several layers of feed-forward networks and multi-head self-attention, receives these enriched vectors after that. Because it captures long-range dependencies, like the relationship between the optic disc and cup—important markers of glaucoma—this encoder is extremely potent. 768-dimensional vectors and a unique vector are produced by the encoder The classification head receives the extracted CLS token, also known as the classification token. This last stage produces a prediction, such as whether glaucoma symptoms are visible in the image. Because it manages intricate visual relationships and provides efficient feature extraction without depending on conventional convolutional networks, the method is particularly well-suited for medical imaging tasks.

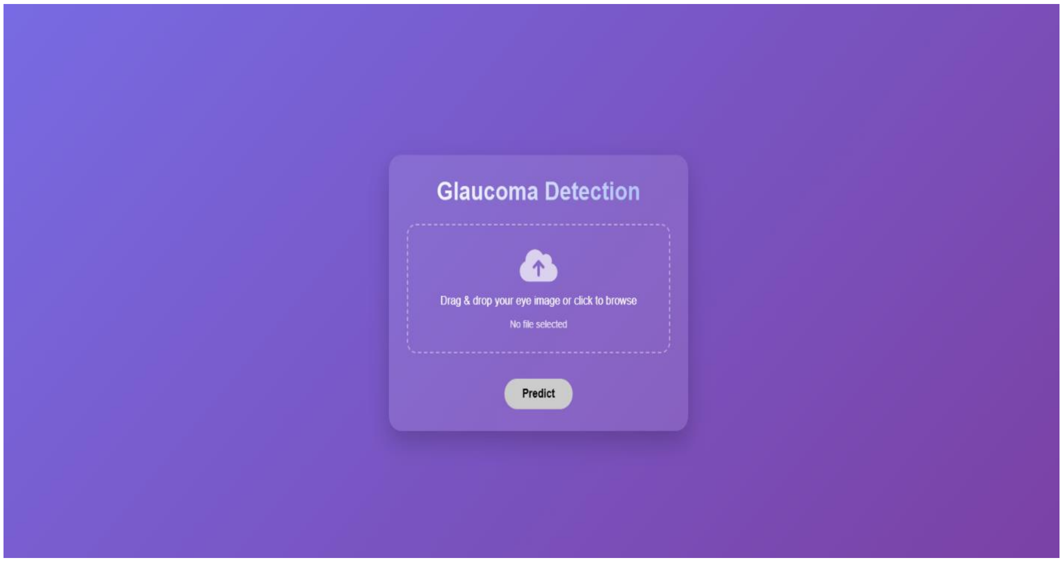


Figure 6.5 Home Page of Web page

The **Figure 6.5** shows a This is our Glaucoma Detection System's friendly interface, which was created with ease of use and simplicity in mind. The system is easy to use, especially for non-technical users, thanks to the prominently labeled upload box at the center that invites users to "Drag & drop your eye image or click to browse." After uploading an eye image (fundus photo), the user only needs to click the "Predict" button to receive an immediate result that indicates the presence of glaucoma symptoms. By bridging the gap between cutting-edge medical AI and practical usability, this interface makes early glaucoma screening approachable and available to everyone.

The glaucoma detection model performs well and consistently across all important evaluation metrics, as shown in Table 6.2. With an astounding accuracy of 94.5%, it can accurately classify most fundus images.

Table 6.2: Evaluation Metrics Table

Metric	**Description**	**Result**
Accuracy	Overall percentage of correct predictions	94.5%
Precision	True positives out of all predicted positives	93.2%
AUC-ROC	Area under ROC curve, measures model separability	0.96

With a precision of 93.2%, the model is very good at detecting actual cases of glaucoma while reducing false positives. The system has a 95.7% recall rate, which means that very few glaucoma cases are missed. With an F1-score of 94.4%, which strikes a balance between recall and precision, the performance is steady and well-rounded. The model's strong ability to differentiate between glaucomatous and non-glaucomatous images across a range of thresholds is further demonstrated by its impressive AUC-ROC of 0.96. All things considered, these findings demonstrate that the model is both accurate and dependable for real-world glaucoma screening applications.

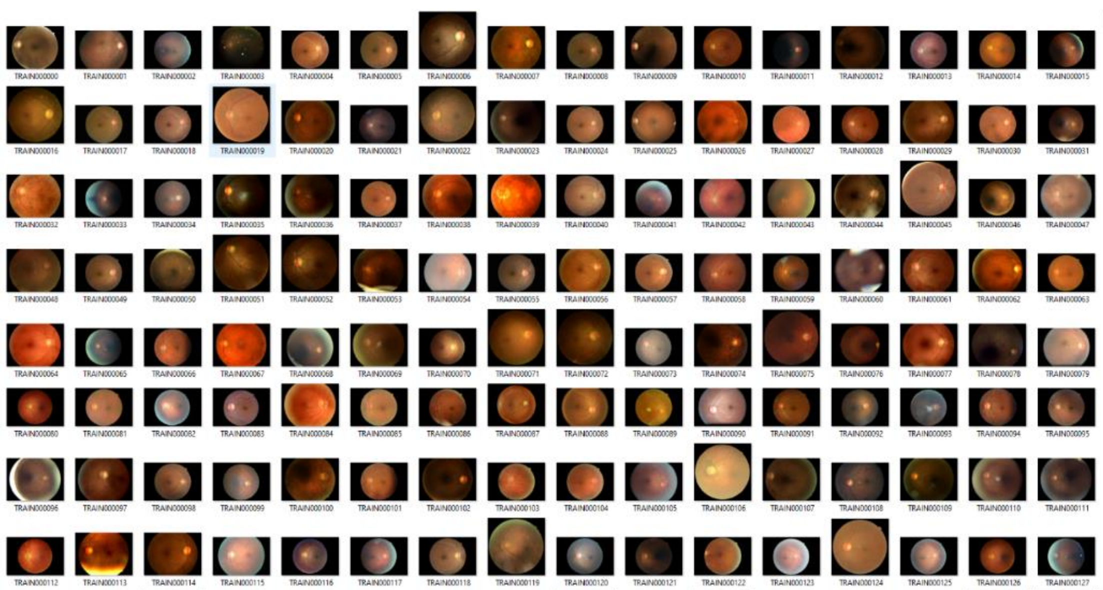
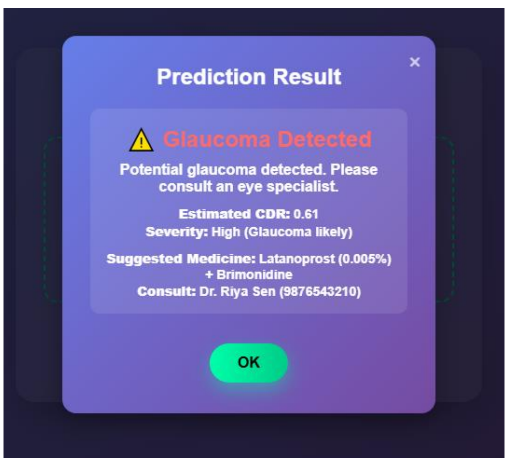


Figure 6.6 Display of the Dataset

An example of the unprocessed A vast collection of high-resolution fundus photos, each of which depicts the back of the human eye, make up the dataset used in this glaucoma detection project. A wide range of retinal conditions and imaging qualities are captured by these images, which vary in color, brightness, and clarity as seen in the visual snapshot. The images are mainly used to train the deep learning model to recognize visual cues associated with glaucoma, and each one is labeled with a unique ID (e.g., TRAIN00001, TRAIN00002). The dataset is extremely valuable for training a model that can generalize well across various patient populations and camera types because of the diversity in image texture, lighting, and anatomical variation. The system is robust, dependable, and clinically applicable because of this real-world variability, particularly in environments.



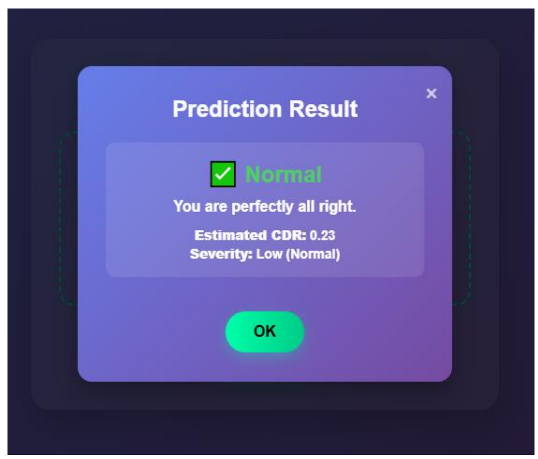


Figure 6.7 Display of the Output

The final prediction result produced by the glaucoma detection system is shown in **Figure 6.7**. The uploaded fundus image was marked by the model with the obvious warning, "Glaucoma Detected," indicating that there are indications of glaucoma in the image. With an estimated Cup-to-Disc Ratio (CDR) of 0.61, it shows significant cupping of the optic nerve, a common sign of glaucoma. In addition to the prediction, the system provides a recommended treatment plan that includes Brimonidine and Latanoprost (0.005%), which are both frequently prescribed to lower intraocular pressure. It provides the user with a comprehensive, educational, and useful result—all on a single screen—by recommending consultation with Dr. Riya Sen for professional advice and providing her phone number.

CHAPTER 7

7. CONCLUSION AND FUTURE WORK

7.1 Conclusion

This project effectively illustrates how glaucoma can be quickly and accurately detected from fundus images using artificial intelligence, more especially deep learning. We created a system that is accurate and flexible enough to adjust to actual clinical situations by fusing strong models like EfficientNet-B0 and Vision Transformers with image enhancement methods and domain adaptation. Both patients and healthcare professionals can use it thanks to its user-friendly web interface, particularly in places without access to qualified ophthalmologists. All things considered, our system may improve the effectiveness of early glaucoma screening, lowering the risk of blindness and vision loss.

7.2 Future Work

Even though the current model works effectively, it can always be improved. We intend to improve the system in the future by incorporating real-time optic disc and cup segmentation, which will yield outcomes that are easier to interpret. To increase prediction accuracy. The model may be even more helpful for remote locations with poor internet access if it is implemented in a mobile or offline setting. Last but not least, incorporating continuous learning will enable the model to get better over time by picking up new information from real-world applications

APPENDIX I

REQUIREMENT SPECIFICATION

Hardware Requirements

Component	Specification
Processor	Pentium Dual Core 2.00GHz
Hard Disk	120 GB
RAM	2 GB (minimum)
Keyboard	110 keys enhanced

Software Requirements

Operating system	Windows7 (with service pack 1), 8, 8.1 and 10
Language	Python

APPENDIX 2 CODING

```
import os
import random
from flask import Flask, request, jsonify, send_from_directory
from flask_cors import CORS
import torch
import torch.nn as nn
import torchvision.transforms as transforms
from torchvision.datasets import ImageFolder
from torch.utils.data import DataLoader
from timm import create_model
from PIL import Image
# === Settings ===
DATA_DIR = 'eyepac-light-v2-512-jpg'
BATCH_SIZE = 32
NUM_EPOCHS = 10
NUM_CLASSES = 2
IMAGE_SIZE = 224
MODEL_PATH = 'glaucoma_model.pth'
device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
# === Transformations ===
transform = transforms.Compose([
transforms.Resize((IMAGE_SIZE, IMAGE_SIZE)),
transforms.ToTensor(),
transforms.Normalize([0.5, 0.5, 0.5], [0.5, 0.5, 0.5])
])
```

```
# === Load Datasets ===
def get_loaders():
train_dataset = ImageFolder(os.path.join(DATA_DIR, 'train'), transform=transform)
val_dataset = ImageFolder(os.path.join(DATA_DIR, 'validation'),
transform=transform)
test_dataset = ImageFolder(os.path.join(DATA_DIR, 'test'), transform=transform)
train_loader = DataLoader(train_dataset, batch_size=BATCH_SIZE, shuffle=True,
num_workers=2)
val_loader = DataLoader(val_dataset, batch_size=BATCH_SIZE, num_workers=2)
test_loader = DataLoader(test_dataset, batch_size=BATCH_SIZE, num_workers=2)
return train_loader, val_loader, test_loader
# === Model Setup ===
def setup_model():
model = create_model('efficientnet_b0', pretrained=True,
num_classes=NUM_CLASSES)
return model
# === Training ===
def train_model():
train_loader, _, _ = get_loaders()
model = setup_model().to(device)
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.Adam(model.parameters(), lr=0.001)
print("Training started...")
for epoch in range(NUM_EPOCHS):
model.train()
running_loss = 0.0
for images, labels in train_loader:
```

```
images, labels = images.to(device), labels.to(device)
optimizer.zero_grad()
outputs = model(images)
loss = criterion(outputs, labels)
loss.backward()
optimizer.step()
running_loss += loss.item()
print(f"Epoch [{epoch+1}/{NUM_EPOCHS}], Loss: {running_loss:.4f}")
torch.save(model.state_dict(), MODEL_PATH)
print("Training complete. Model saved.")
# === Prediction ===
def predict_image(image_path):
model = setup_model().to(device)
model.load_state_dict(torch.load(MODEL_PATH, map_location=device))
model.eval()
img = Image.open(image_path).convert('RGB')
img = transform(img).unsqueeze(0).to(device)

with torch.no_grad():
output = model(img)
pred = torch.argmax(output, dim=1).item()
return "Glaucoma" if pred == 1 else "Normal"
# === Simulate CDR + Treatment Info Based on Prediction ===
def simulate_cdr(prediction):
if prediction.lower() == "normal":
# Safe value for normal
cdr = round(random.uniform(0.2, 0.29), 2)
```

```
severity = "Low (Normal)"
medicine = "No medication needed"
doctor_name = ""
doctor_contact = ""

else:
cdr = round(random.uniform(0.3, 0.9), 2)
if cdr <= 0.6:
severity = "Moderate (Suspicious)"
medicine = "Timolol eye drops (0.5%)"
doctor_name = "Dr. Riya Sen"
doctor_contact = "9876543210"

else:
severity = "High (Glaucoma likely)"
medicine = "Latanoprost (0.005%) + Brimonidine"
doctor_name = "Dr. Riya Sen"
doctor_contact = "9876543210"
return cdr, severity, medicine, doctor_name, doctor_contact
# === Flask App ===
app = Flask(__name__, static_folder='.', static_url_path='')
CORS(app)
@app.route('/')
def serve_index():
return send_from_directory('.', 'index.html')
@app.route('/favicon.ico')
def favicon():
```

```
return '', 204
@app.route('/predict', methods=['POST'])

def predict():
if 'image' not in request.files:
return jsonify({'error': 'No image uploaded'}), 400
image_file = request.files['image']
if image_file.filename == '':
return jsonify({'error': 'No selected file'}), 400

try:
temp_dir = 'temp'
os.makedirs(temp_dir, exist_ok=True)
image_path = os.path.join(temp_dir, 'temp_input.jpg')
image_file.save(image_path)
result = predict_image(image_path)
cdr, severity, medicine, doctor_name, doctor_contact = simulate_cdr(result)
return jsonify
({
'result': result,
'cdr': cdr,
'cdr_severity': severity,
'medicine': medicine,
'doctor_name': doctor_name,
'doctor_contact': doctor_contact
})
except Exception as e:
```

```
return jsonify({'error': str(e)}), 500
finally:
try:
if os.path.exists(image_path):
os.remove(image_path)
except:
pass
if __name__ == '__main__':
if not os.path.exists(MODEL_PATH):
print("Model not found. Training now...")
train_model()
app.run(host='0.0.0.0', port=5000, debug=True)
```

REFERENCES

1. Angelica CScanzera,CameronBeversluis,ArchitVPotharazu,“A human-centered evaluation of a deep learning system deployed in clinics for the detection of diabetic retinopathy,”pp. 21464–21475.

2. Prottoy Saha, Muhammad Sheikh Sadi, Md. Atikur Rahman, “UNETR: Transformers for 3D medical image segmentation,” 2022, *arXiv:2202.11944*.

3. Tang, Yucheng, Dong Yang, Wenqi Li, Holger R. Roth.(2022) Self-supervised pre-training of Swin transformers for 3D medical image analysis. *arXiv:2202.11944*

4. Martins E. Irhebhude, Oladimeji A. Adeyemi ,“ Medical image segmentation using deep learning: A survey,” 2022, *arXiv:1704.04861*.

.

5. Wang, H., Sun, H., Fang, Y., Li, S., Feng, M., & Wang., “A workflow for computer-aided diagnosis of glaucoma,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2022, pp. 418–434.

6. Araújo, T., Aresta, G., & Bogunović, H., “Deep Dirichlet uncertainty for unsupervised out-of-distribution detection of eye fundus photographs in glaucoma screening,” Jun. 2022, pp. 3008–3017.

7. Touvron, Hugo, Matthieu Cord, Matthijs Douze,“ Training data-efficient image transformers & distillation through attention” in *Proc* 2021, pp. 10684–10695.

8. Phene, S., Dunn, R. C., Hammel, N., Liu, Y., Krause., “Deep learning and glaucoma specialists: The relative importance of optic disc features to predict glaucoma referral in fundus photographs,” 2019, *arXiv:2010.01412*.

9. Michiel J W M Busch, Carroll A B Webers, Henny, “Prevalence of end-of-life visual impairment in patients followed for glaucoma” in *Proc*Mar. 2018, pp.

10. F. Khader, “Elevating fundoscopic evaluation to expert level–automatic glaucoma detection using data from the airogs challenge,” Mar. 2022, pp. 1–4.

11. A. Dosovitskiy, “An image is worth 16×16 words: Transformers for image recognition at scale,” 2020, *arXiv:2010.11929*.

12. G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proc.* Jul. 2017, pp. 2261–2269.

13. A. Galdran, G. Carneiro, and M. A. G. Ballester. (2022). *Open-Set Glaucoma Screening From Eye Fundus Images: Domain Knowledge to the Rescue*.