SEGMENTATION AND CLASSIFICATION OF DIABETIC RETINOPATHY

A PROJECT REPORT

Submitted by

Sayan Kumar Bag [Reg No:RA2011003010951]

Shivansh Singh [Reg No: RA2011003010950]

Under the Guidance of

Dr. SANKARA NARAYANAN S

Assistant Professor, Department of Computing Technologies

in partial fulfilment of the requirements for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTING TECHNOLOGIES

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR– 603 203

NOVEMBER 2023

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR–603 203

BONAFIDE CERTIFICATE

Certified that the 18CSP107L project report titled " **SEGMENTATION AND CLASSIFICATION OF DIABETIC RETINOPATHY** " is the bonafide work of **SAYAN KUMAR BAG [RegNo:RA2011003010951]** and **SHIVANSH SINGH [RegNo:RA2011003010950]** who carried out the project work under my supervision. To the best of my knowledge, the work reported here does not form part of any other thesis or dissertation based on which a degree or award was conferred on an earlier occasion for this or any other candidate.

Dr. SANKARA NARAYANAN S
SUPERVISOR
Assistant Professor
Department of Computing Technologies

Dr. JOTHI KUMAR C
PANEL HEAD
Associate Professor
Department of Computing Technologies

Dr M. PUSHPALATHA
HEAD OF THE DEPARTMENT
Department of Computing Technologies

Department of Computing Technologies

SRM Institute of Science and Technology

Own Work Declaration Form

Degree/Course	: B.Tech in Computer Science and Engineering
Student Names	: SAYAN KUMAR BAG, SHIVANSH SINGH
Registration Number	: RA2011003010951, RA2011003010950
Title of Work	:SEGMENTATION AND CLASSIFICATION OF DIABETIC RETINOPATHY

I/We hereby certify that this assessment compiles with the University's Rules and Regulations relating to Academic misconduct and plagiarism, as listed in the University Website, Regulations, and the Education Committee guidelines.

I / We confirm that all the work contained in this assessment is our own except where indicated, and that we have met the following conditions:

- references / listed all sources as appropriate
- Referenced and put in inverted commas all quoted text (from books, web, etc.)
- Given the sources of all pictures, data etc that are not my own.
- Not made any use of the report(s) or essay(s) of any other student(s)either past or present
- Acknowledged in appropriate places any help that I have received from others (e.g fellow students, technicians, statisticians, external sources)
- Compiled with any other plagiarism criteria specified in the Course hand book / University website

I understand that any false claim for this work will be penalized in accordance with the University policies and regulations.

DECLARATION:
I am aware of and understand the University's policy on Academic misconduct and plagiarism and I certify that this assessment is my / our own work, except where indicated by referring, and that I have followed the good academic practices noted above. **Student 1 Signature:** **Student 2 Signature:** **Date:**
If you are working in a group, please write your registration numbers and sign with the date for every student in your group.

ACKNOWLEDGEMENT

We express our humble gratitude to **Dr. C. Muthamizhchelvan**, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support.

We extend our sincere thanks to Dean-CET, SRM Institute of Science and Technology, **Dr. T. V. Gopal**, for his invaluable support.

We wish to thank **Dr. Revathi Venkataraman**, Professor and Chairperson, School of Computing, SRM Institute of Science and Technology, for her support throughout the project work.

We are incredibly grateful to our Head of the Department, **Dr. M. Pushpalatha,** Professor, Department of Computing Technologies, SRM Institute of Science and Technology, for her suggestions and encouragement at all stages of the project work.

We want to convey our thanks to our Project Coordinators, **Dr. S. Godfrey Winster,** Panel Head, **Dr. Jothi Kumar C**, Associate Professor and Panel Members, **Dr. Deepa Jothi S** Assistant Professor, **Dr. C. AshokKumar** Assistant Professor and **Dr. Sankar Narayanan S** Assistant Professor, Department of Computing Technologies, SRM Institute of Science and Technology, for their inputs during the project reviews and support.

We register our immeasurable thanks to our Faculty Advisor, **Dr. Viji D**, Assistant Professor, Department of Computing Technologies, SRM Institute of Science and Technology, for leading and helping us to complete our course.

Our inexpressible respect and thanks to our guide, **Dr. Sankara Narayanan S**, Assistant Professor, Department of Computing Technologies, SRM Institute of Science and Technology, for providing us with an opportunity to pursue our project under his mentorship. He provided us with the freedom and support to explore the research topics of our interest. His passion for solving problems and making a difference in the world has always been inspiring.

We sincerely thank all the staff and students of the Computing Technologies Department, School of Computing, S.R.M Institute of Science and Technology, for their help during our project. Finally, we would like to thank our parents, family members, and friends for their unconditional love, constant support and encouragement.

SAYAN KUMAR BAG [RA2011003010951]
SHIVANSH SINGH [RA2011003010950]

ABSTRACT

In this project, we address the critical issue of precise Diabetic Retinopathy (DR) diagnosis, a condition often leading to severe vision impairment or blindness in diabetic patients. Leveraging advanced deep learning models and innovative image processing techniques, our study focuses on accurate retinal image segmentation using a U-Net model. This segmentation method delineates lesions and retinal structures effectively. Subsequently, Gabor filters are applied for intricate texture pattern extraction, indicative of diverse retinopathy stages. Integrating MobileNetV2 for feature extraction and EfficientNetB0 for multi-class classification significantly enhances the diagnostic accuracy. Our developed system exhibits a promising 60.10% test accuracy, showcasing its potential in DR diagnosis. While challenges related to varying severity levels persist, our robust framework lays the groundwork for future refinements. By amalgamating sophisticated image segmentation, feature extraction, and classification techniques, our system provides a solid foundation for accurate and timely DR assessment. With continuous enhancements, including the incorporation of more extensive and diverse datasets, our approach holds the promise to revolutionise DR diagnostics. The integration of cutting-edge technology into medical practices underscores the transformative impact of artificial intelligence in the realm of ophthalmology, promising improved patient outcomes.

TABLE OF CONTENTS

	ABSTRACT		**5**
	LIST OF TABLES		**8**
	LIST OF FIGURES		**9**
	LIST OF SYMBOLS AND ABBREVIATIONS		**10**
1.	**INTRODUCTION**		**11**
	1.1	General	11
	1.2	Machine Learning & Deep Learning	12
	1.3	Difference Between Machine Learning & Deep Learning	13
	1.4	Deep Learning	14
	1.5	Deep Learning in Healthcare	15
	1.6	Convolutional Neural Network	16
	1.7	U-Net	18
	1.8	EfficientNet	20
2	**LITERATURE SURVEY**		**21**
	2.1	Motivation	21
	2.2	Objective	24
3	**ARCHITECTURE AND ANALYSIS**		**25**
	3.1	Architecture Diagram	25
	3.2	Architecture Analysis	27
4	**DESIGN AND IMPLEMENTATION**		**29**
	4.1	Dataset	29
	4.2	Image Preprocessing	31
	4.3	Feature Extraction	32
	4.4	Image Segmentation	33
	4.5	Classification	36
		4.5.1 EfficientNet model	37

5	**RESULTS AND DISCUSSION**	**39**
	5.1 Performance Analysis using Various Metrics	39
	5.2 Comparison Between Existing Models	42
6	**CONCLUSION AND FUTURE SCOPE**	**44**
	6.1 Conclusion	44
	6.2 Future Scope	45
	REFERENCES	**46**
	APPENDIX	**48**
	PLAGIARISM REPORT	**56**

LIST OF TABLES

5.1	Classification Performance of the Model	43

LIST OF FIGURES

1.1	CNN Architecture	16
1.2	U-Net Architecture	18
1.3	EfficientNetB0 Architecture	20
3.1	Architecture Diagram of Proposed Model	25
4.1	Aptos 2019 Dataset	29
4.2	Balanced Dataset	30
5.1	Test Accuracy	39
5.2	Classification Report	39
5.3	Confusion Matrix	40
5.4	Graph of Training and Validation Accuracy	41

LIST OF SYMBOLS AND ABBREVIATIONS

DR	**Diabetic Retinopathy**
CNN	**Convolutional Neural Network**
AI	**Artificial Intelligence**
NLP	**Natural Language Processing**
RNN	**Recurrent Neural Networks**

CHAPTER -1

INTRODUCTION

1.1 General

Diabetic Retinopathy (DR) stands as a significant public health concern, threatening the eyesight of millions worldwide [1]. It is a diabetes-related complication leading to retinal damage and vision impairment, making early and accurate diagnosis imperative for effective management [2]. Recent advancements in artificial intelligence, specifically deep learning models and innovative image processing techniques, have shown promising results in analysis of medical images , particularly in ophthalmology [3] [4]. This project delves into the development of an advanced diagnostic system for DR, combining cutting-edge technologies to enhance the efficiency and precision of the diagnosis process.

Research community has witnessed remarkable progress in the application of deep learning models, such as the UNet architecture, in medical image segmentation [5]. This project leverages UNet for precise retinal image segmentation, enabling the delineation of lesions and retinal structures. Moreover, the integration of Gabor filters provides a robust approach for capturing intricate texture patterns indicative of various retinopathy stages [6].

In addition to these segmentation and feature extraction techniques, the study incorporates pre-trained convolutional neural networks (CNNs) like MobileNetV2 for efficient feature extraction and EfficientNetB0 for multi-class classification [7] [8]. The utilisation of these state-of-the-art models significantly enhances the diagnostic accuracy, laying the foundation for a reliable DR diagnostic system.

This project aims not only to contribute to the evolving landscape of AI-driven medical diagnostics but also to address the pressing need for accurate and timely DR assessment. By amalgamating these sophisticated technologies, this research seeks to pave the way for a transformative impact on ophthalmic healthcare, ultimately improving patient results and decreasing the burden of DR-related vision loss.

1.2 Machine Learning & Deep Learning

Two major areas of AI, Machine Learning and Deep Learning, revolutionized how machines could learn and make decisions. Machine learning algorithms help computers to find patterns in the data, and predict or make decisions that are not explicitly programmed. In particular, in cases where traditional Rule Based Programming methods may not be feasible, this approach has been particularly useful to address huge and complicated data sets.

Deep learning is a form of machine learning that looks at neural networks that have many numbers of layers, which is why it's called "deep". These deep neural networks can learn difficult patterns from huge amounts of data, which makes them really useful for things like recognizing things like images and speech, understanding natural language, and playing games. Basically, deep learning models are inspired by the way the human brain works, with connected neurons that process and change information.

Deep learning offers a significant benefit: it can extract important features from raw data without requiring manual feature engineering. This advantage is especially valuable in tasks involving high-dimensional input data, like images and audio signals. Impressive outcomes have been achieved in real-world applications by deep learning models, with convolutional neural networks (CNNs) for pictures and recurrent neural networks (RNNs) for sequences playing a particularly noteworthy role. These applications span a wide range, including medical diagnosis, autonomous vehicles, language translation, and virtual assistants.

The strengths and weaknesses of machine learning & deep learning vary, and the decision of which to use depends on the problem, data availability, and computational resources. As technology progresses, these areas will have a significant impact on the future of AI, making once fictional innovations a part of our daily lives.

1.3 Difference between Machine Learning & Deep Learning

Machine learning and deep learning are two branches of artificial intelligence (AI) that have the common goal of teaching machines to learn from data and make informed decisions. Despite their similarities, they differ in their methods, structures, and the types of problems they are most effective at solving.

Machine Learning:
Machine learning is a comprehensive concept that encompasses a range of algorithms and techniques designed to enhance a computer's ability to learn from data and improve its performance over time. In this field, algorithms are trained using labeled data, allowing them to detect patterns, make predictions, and take actions based on their training. Different types of machine learning algorithms exist, including supervised, unsupervised, semi-supervised, and reinforcement learning, each suited to specific tasks.

Supervised Learning:
Supervised learning includes training algorithms with labeled data, where the input data is paired along with corresponding output labels. Through this training, the algorithm learns to associate input data with the correct output labels, enabling it to make accurate predictions when presented with new, unseen data.

Unsupervised Learning:
On the other hand, unsupervised learning revolves around training algorithms using unlabeled data, with the aim of uncovering patterns, clusters, or hidden structures within the data. Popular techniques used in unsupervised learning include clustering and dimensionality reduction.

Interacting with an environment, an agent makes decisions and learns to maximize rewards over time, making it suitable for decision-making tasks. Deep learning, a part of machine learning on neural networks with multiple layers. These networks, inspired by the human brain, can learn to extract complex patterns from large amounts of data. Convolutional neural networks (CNNs) for images and recurrent neural networks (RNNs) for sequences are well-suited for high-dimensional data tasks like images, speech, and natural language. The main differences between machine learning and deep learning lie in the complexity of models and the automatic feature extraction capabilities of deep learning architectures. In traditional machine learning, feature engineering is a manual and crucial step, requiring expertise. However, deep learning models can spontaneously learn important features from input data, making them highly effective for complex datasets.

Deep learning is a dominant approach that utilizes neural networks with multiple layers to automatically extract features and learn representations from unprocessed data. Its impressive achievements in tasks like identifying images, processing speech, and understanding natural language have made it a preferred option for AI applications that demand advanced pattern recognition abilities.

1.4 Deep Learning

Deep learning, a part of machine learning, has gained immense popularity for its power to train complex artificial neural networks. These networks, with multiple layers, automatically extract intricate complex patterns and features from input data. The depth of these neural networks enables them to learn multiple levels of abstraction, allowing for sophisticated data representation. An essential advantage of deep learning is its automatic feature extraction capability, eliminating the need for manual feature engineering, especially beneficial for high-dimensional data like images and text.

Deep learning has significantly impacted various fields, including computer vision, NLP, and speech recognition. For instance, convolutional neural networks (CNNs) in computer vision can recognize detailed patterns, making them invaluable in tasks like object detection and facial recognition. In natural language processing, recurrent neural networks (RNNs) and transformer models have revolutionized machine translation and language generation. Despite its successes, deep learning faces challenges. Training deep neural networks demands substantial computational resources, and overfitting issues require careful handling through techniques like regularization.

Additionally, deep learning models' interpretability remains a challenge, limiting their applications in sensitive domains. Despite these challenges, ongoing research in deep learning continues to expand its capabilities, pushing the boundaries of artificial intelligence. Its applications extend to diverse areas such as autonomous vehicles, healthcare diagnostics, and personalized recommendation systems. As the field evolves, addressing these limitations is crucial for unlocking the full potential of deep learning and ensuring its broader integration into real-world applications.

1.5 Deep Learning in Healthcare

Healthcare has been completely transformed by the emergence of deep learning, a technology that has revolutionized various aspects of medicine. Its impact ranges from improving disease diagnosis and medical imaging to enhancing drug discovery and creating personalized treatment plans. One of the key advantages of deep learning in healthcare is its ability to rapidly and accurately process and analyze extensive amounts of medical data, leading to better patient outcomes and more efficient healthcare processes.

In the realm of medical imaging, deep learning algorithms, including convolutional neural networks (CNNs), have demonstrated exceptional performance in tasks such as tumor detection, identifying abnormalities in X-rays and MRI scans, and segmenting organs. These algorithms can learn intricate patterns and subtle nuances in images, assisting radiologists in making more precise diagnoses. This not only reduces the time required for analysis but also improves the early detection of diseases, increasing the likelihood of successful treatment.

Deep learning also plays a critical role in genomics and drug discovery. By analyzing genetic data, deep learning models can identify potential biomarkers, predict disease risks, and optimize drug combinations based on specific genetic profiles. This personalized approach to medicine has the potential to revolutionize treatment strategies, making them more targeted and effective, while minimizing the occurrence of adverse effects.

Deep learning plays a crucial role in extracting valuable insights from nonstructured medical data, like doctors' notes, research papers, and records of patients . This enables researchers to mine data for their studies, helps healthcare providers make faster clinical decisions, and enhances their understanding of patients' medical histories, ultimately leading to better patient care.

However, incorporating deep learning in healthcare does pose some challenges. These include the need for big and diverse datasets, ensuring the privacy and security of data, and addressing regulatory compliance issues. Despite these obstacles, the continuous advancements in deep learning algorithms and the increasing availability of healthcare data hold great potential for revolutionizing healthcare delivery. This transformation would result in more precise, efficient, and patient-centered care. As research and development in this field progress, the integration of deep learning technologies is expected to impact a vital role in shaping the future of healthcare.

1.6 Convolutional Neural Network

A specialized form of artificial neural network, a Convolutional Neural Network (CNN) is designed specifically for processing data arranged in a grid-like format, such as pictures and videos. CNNs are especially effective in tasks related to computer vision, which involve identifying patterns, objects, or characteristics within visual data.

CNNs possess a unique capability to automatically and flexibly learn spatial graph of features from input images. This is accomplished by utilizing convolutional layers, which apply convolution operations to the input data. Convolution entails sliding a small window, referred to as a kernel or filter, across the input image. As the filter covers a specific local region of the image, element-wise multiplication is performed between the filter and the picture region, and the results are then summed up. Through this process, the network becomes capable of capturing diverse features at different spatial scales. Consequently, CNNs are well-suited for tasks that require hierarchical feature learning.

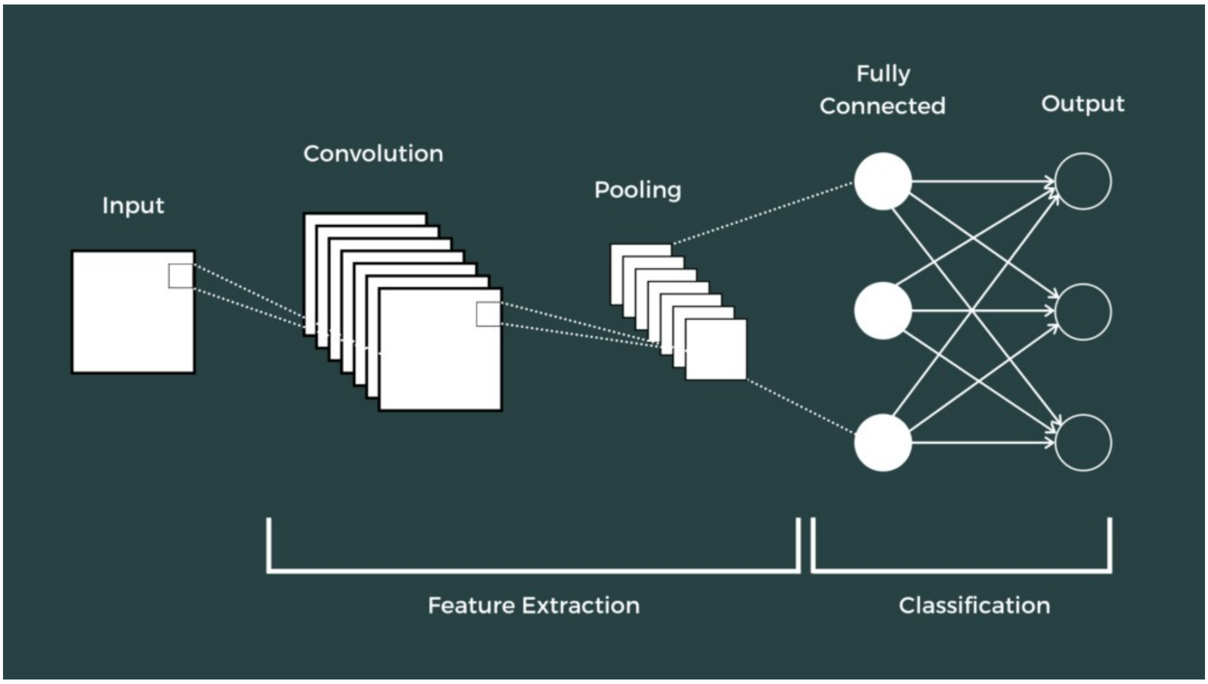


Fig.1.1- CNN Architecture

Pooling layers are commonly incorporated in CNNs to reduce the size of the input volume. This helps to simplify the network's computations and improve its capability to detect patterns across different positions in an image, even when there are variations in scale and translation.

Typical CNN architectures consist of multiple convolutional layers, which are followed by activation functions like ReLU, and pooling layers. The outputs from these layers are then flattened and passed through fully connected layers, ultimately producing the final output, such as class scores for a classification task. During training, CNNs often use labeled data and optimization techniques like backpropagation and gradient descent to minimize the disparity between predicted outputs and actual labels.

CNNs have exhibited impressive performance in a many of applications, like image classification, object detection, face recognition, and image generation. Their ability to autonomously learn important features from raw pixel values has established them as a crucial component of modern computer vision systems. This, in turn, has enabled advancements in fields such as autonomous vehicles, medical imaging, and augmented reality. As a result, CNNs have become a foundational technology in the realm of artificial intelligence, driving numerous real-world applications that heavily rely on the analysis of visual data.

1.7 U-Net

U-Net is a convolutional neural network (CNN) and was created to address semantic segmentation tasks, which entail categorizing every pixel in an image into distinct classes. U-Net has gained significant popularity in the realm of medical image analysis, particularly in tasks like organ detection, cell segmentation, and tumor delineation. Its name is derived from its unique architecture, which resembles the shape of the letter U, consisting of a contracting path followed by an expansive path.

The U-Net architecture addresses a common issue in semantic segmentation: the challenge of capturing both local and global context information. Local information is essential for precise pixel-wise predictions, while global context helps in understanding the overall image context. U-Net achieves this by employing a symmetric, fully convolutional network with skip connections.

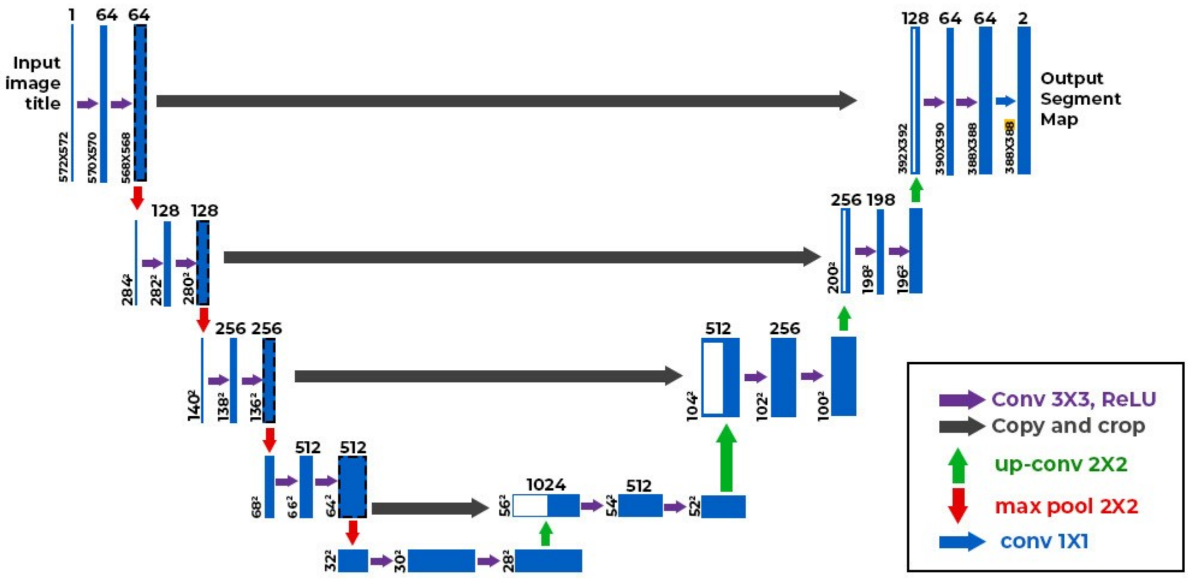


Fig. 1.2 - U-Net Architecture

Consisting of three major components, the U-Net structure encompasses the contracting path, the bottleneck, and the expansive path. The contracting path, or the encoder, utilizes convolutional and max-pooling layers to gather contextual information. By progressively decreasing the spatial dimensions of the input image and simultaneously increasing the feature channels, this path enables the network to acquire hierarchical features spanning from local to global scales.

At the heart of the U-Net lies the bottleneck, acting as a crucial link between the contracting and expansive paths. Within this section, multiple convolutional layers extract intricate features from the encoded information. Notably, skip connections are introduced here, connecting corresponding layers from the contracting path to the expansive path. These skip connections play a vital role in preserving high-resolution spatial information and capturing fine details, effectively mitigating the loss of spatial accuracy during downsampling.

The expansive path, also known as the decoder, restores the feature maps to their original input image resolution. This is achieved through the use of transposed convolutions, which increase the spatial dimensions. Simultaneously, skip connections concatenate the high-resolution feature maps from the contracting path, allowing the network to recover spatial details that may have been lost in the contracting path. The final layer typically employs a convolution operation with a softmax activation function to generate pixel-wise predictions. These predictions give rise to segmentation masks for each class of interest.

One of the standout advantages of the U-Net is its ability to handle limited labeled data. This is made possible through the implementation of data augmentation techniques, such as rotation, flipping, and scaling. By applying these techniques, a small dataset can be transformed into a diverse set of training samples.

By starting the network with pre-existing weights from a similar task or dataset, one can improve the model's performance, particularly in situations where there is a lack of training data.

U-Net's versatility and effectiveness have led to its widespread adoption in medical image analysis and beyond. Its applications extend to tasks like image-to-image translation, image denoising, and even non-image data such as 3D point cloud segmentation. The U-Net architecture continues to inspire further research, fostering innovations in semantic segmentation and contributing to advancements in computer vision and deep learning.

1.8 EfficientNet

EfficientNet, a revolutionary convolutional neural network architecture, aims to achieve exceptional performance while utilizing fewer parameters and computational resources compared to earlier models. Created by Google's Mingxing Tan and Quoc V. Le, EfficientNet presents a fresh perspective on scaling neural networks by prioritizing the equilibrium between model size and accuracy. This groundbreaking design has garnered good popularity in computer vision works such as image classification, object detection, and segmentation.

The key insight behind EfficientNet is the concept of compound scaling, which optimally balances three critical dimensions of a neural network: depth (total no. of layers), width (no. of channels in each layer), and resolution (input image size). Unlike traditional methods that independently scale these dimensions, EfficientNet employs a compound scaling coefficient to uniformly scale all three aspects. This approach ensures that each dimension is expanded judiciously, leading to improved performance without significantly increasing the computational cost.

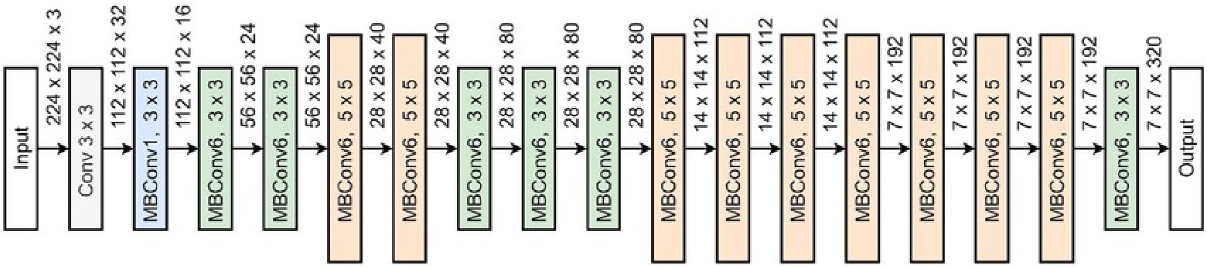


Fig. 1.3 -EfficientNetB0 Architecture

EfficientNet architectures are denoted as EfficientNet-B0 to EfficientNet-B7, with each variant being progressively larger and more powerful. EfficientNet-B0 serves as the baseline model, and subsequent versions (B1 to B7) are scaled up versions of the baseline, achieving better accuracy while maintaining computational efficiency. The scaling coefficients are determined through a principled method that optimally balances the trade-off between precision and computational cost. This method involves finding the right balance between the no. of layers, the no. of channels, and the input resolution.

EfficientNet has achieved remarkable success by employing compound scaling to deliver exceptional performance in various tasks and datasets. Through optimal resource utilization and maximizing the model's representational capacity, EfficientNet has revolutionized the computer vision field by setting unprecedented benchmarks for accuracy and efficiency. Its ability to maintain accuracy while being lightweight makes it especially valuable for deployment in environments with limited computational resources, such as mobile and edge devices.

EfficientNet has shown to be highly precise in image classification tasks, surpassing larger and more computationally demanding models on well-known datasets like ImageNet. Its streamlined design enables real-time deployment, facilitating swift and precise image recognition across multiple platforms. Moreover, EfficientNet has showcased impressive outcomes in various computer vision undertakings, such as object detection, semantic segmentation, and facial recognition.

The impact of EfficientNet extends far beyond the realms of academia and research. Its practical efficiency and exceptional performance have made it a favored choice among both practitioners and researchers. By establishing a new standard for balancing model complexity and accuracy, EfficientNet has profoundly influenced the design principles of neural network architectures. Consequently, this has led to the emergence of more efficient and powerful models in the domain of deep learning.

In essence, EfficientNet's groundbreaking approach to compound scaling has revolutionized the landscape of neural network architectures. Its ability to achieve cutting-edge performance while optimizing resource utilization has positioned it as a pivotal technology in the field of computer vision. As research in this field progresses, the principles of EfficientNet are bound to inspire further innovations, pushing the boundaries of what can be achieved in the development of efficient and accurate deep learning models.

CHAPTER -2

LITERATURE SURVEY

2.1 Motivation

The field of automated diagnosis for Diabetic Retinopathy (DR) has experienced a rise in research efforts, utilizing advanced technologies to improve the precision and effectiveness of diagnostic procedures. Many investigations have delved into the need of deep learning models, inventive image processing methods, and convolutional neural networks (CNNs) in the realm of eye healthcare.

Researchers like Gulshan et al. [9] pioneered the use of deep learning algorithms for the detection of DR in retinal fundus photographs, demonstrating the potential of neural networks in automating the diagnostic process. Following this, Ting et al. [10] extended the application of deep learning to diverse populations with diabetes, showcasing the effectiveness of these models in detecting various stages of DR and related eye diseases.

In the realm of image segmentation, Ronneberger et al. [11] introduced the U-Net architecture, a revolutionary approach for biomedical image segmentation. This architecture has been greatly adopted in medical image analysis tasks, including retinal images segmentation for DR diagnosis. Moreover, Manjunath et al. [12] pioneered the use of texture features for browsing and retrieving image data, providing essential insights into the texture analysis techniques used in DR diagnosis.

In terms of model architectures, Sandler et al. [13] introduced MobileNetV2, emphasising the importance of efficiency in deep learning models. This lightweight architecture is particularly suited for resource-constrained applications, ensuring swift and efficient processing of retinal images. Additionally, Tan et al. [14] proposed EfficientNet, a model scaling technique redefining the standard for CNN architectures. Its superior performance in image recognition has made it a cornerstone in various medical image analysis applications, including DR diagnosis.

Other notable contributions include the work of Abràmoff et al. [15] and Abràmoff et al. [16], which introduced AI-based systems for diabetic retinopathy screening, emphasising the importance of timely and accurate diagnosis in preventing vision loss. Additionally, the study by Ting et al. [17] further refined AI-driven diagnostic tools, showcasing the significance of continuous improvements in model performance and robustness.

Furthermore, the research done by Li et al. [18] and Wang et al. [19] explored innovative feature extraction techniques, including wavelet transform and morphological operations, enriching the toolkit of methodologies used in DR diagnosis. Similarly, the work of Rajalakshmi et al. [20] highlighted the potential of teleophthalmology and AI-driven diagnosis in reaching underserved populations, expanding the scope of DR diagnostic solutions.

In summary, these pioneering studies collectively form the foundation for the current research, providing valuable insights into deep learning applied models, segmentation techniques of images , and innovative feature extraction methods in the realm of Diabetic Retinopathy diagnosis.

2.2 Objective

The main goal of this project is to improve the accuracy and effectiveness of diagnosing diabetic retinopathy by utilizing advanced deep learning techniques. Our objective is to create a strong classification model that can accurately categorize retinal images into five stages: No Diabetic Retinopathy (No DR), Mild Diabetic Retinopathy (Mild DR), Moderate Diabetic Retinopathy (Moderate DR), Severe Diabetic Retinopathy (Severe DR), and Proliferative Diabetic Retinopathy (Proliferative DR). This will be achieved by employing cutting-edge Convolutional Neural Networks (CNNs) and innovative image processing methods.

In addition, we will explore the potential of pre-trained models and transfer learning techniques to optimize the efficiency of our CNN architecture. By leveraging knowledge from existing models, we aim to enhance the model's ability to extract intricate features from retinal images, leading to more accurate classification. Furthermore, the project will implement advanced image segmentation techniques, such as UNet and thresholding, to precisely identify regions of interest within the retinal images. This step is crucial for isolating relevant features and improving the diagnostic accuracy of the model.
Another objective is to utilize feature extraction methods, particularly Gabor filters, to capture complex patterns and textures present in the retinal images.

The primary focus of this project is to differentiate between various stages of diabetic retinopathy. Our goal is to accurately evaluate the model's performance by utilizing comprehensive metrics such as precision, recall, and F1-score. These metrics will help us identify the model's strengths and areas that need improvement, allowing us to refine and optimize it through an iterative process.

In addition, this project aims to make significant contributions to the field of medical image analysis by exploring innovative approaches to diagnose diabetic retinopathy. Our objective is to develop a classification model that is highly accurate and reliable, enabling early and precise detection of this condition. By achieving this, we hope to improve patient outcomes. Ultimately, our project aligns with the broader objective of leveraging technology to advance healthcare, highlighting the transformative impact of deep learning in medical diagnosis and patient care.

CHAPTER -3

ARCHITECTURE AND ANALYSIS

3.1 Architecture Diagram

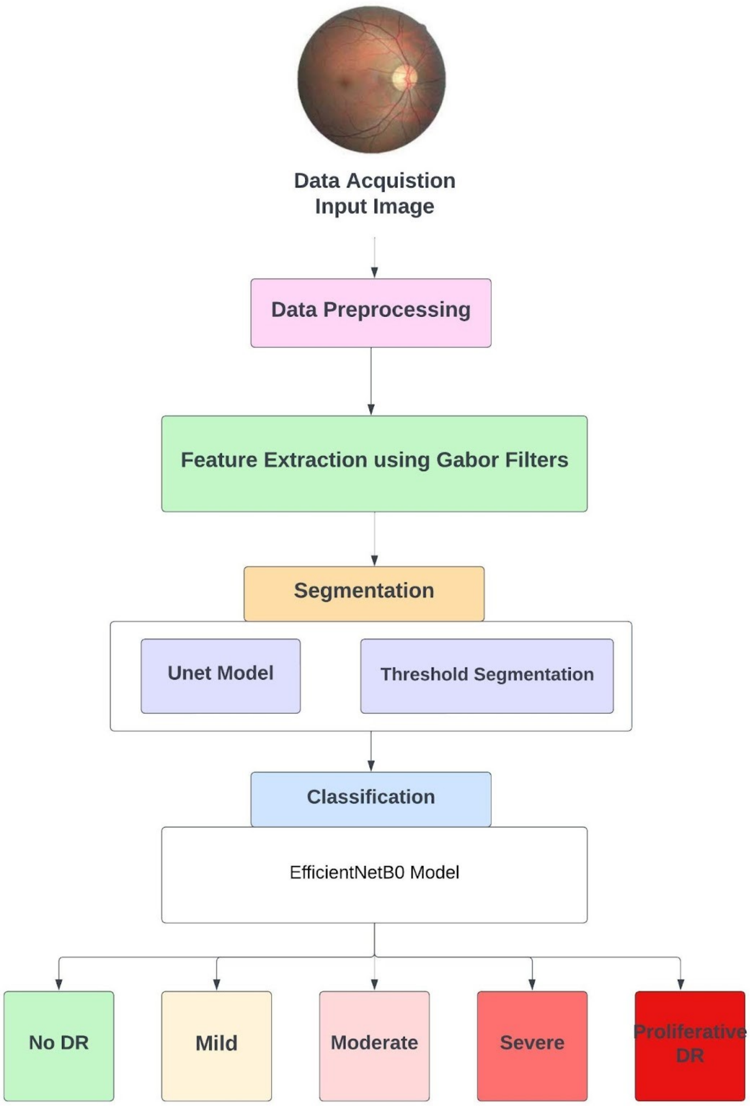


Fig. 3.1- Architecture Diagram of Proposed Model

The suggested design starts by obtaining input images, which form the basis of the analysis. To ensure consistency and facilitate analysis, these unprocessed images undergo a sequence of preprocessing steps. The preprocessing phase includes adjusting the images to a standardized size, normalizing the pixel values, and converting labels into a suitable format for classification.

Following preprocessing, the images are subjected to segmentation techniques, starting with a UNet-based approach. UNet, a deep learning architecture specifically designed for semantic segmentation, aids in identifying relevant regions within the retinal images. This step is crucial as it allows system to focus on the specific areas of interest, enhancing the accuracy of subsequent analyses. Additionally, a thresholding segmentation method is applied, further refining the segmented regions and facilitating the isolation of distinct features.

The next stage involves feature extraction, where Gabor filters are utilized to capture intricate texture patterns within the segmented regions. Gabor filters are well-suited for extracting complex textures and are particularly valuable in medical image analysis. By applying these filters, the system gains a deeper understanding of the subtle details present in the retinal images, enhancing the overall feature representation.

Upon extracting meaningful features, the processed data is fed into an EfficientNet-based classification model. EfficientNet, known for its efficiency and effectiveness in various tasks, is employed to categorize the retinal images into five distinct classes: No DR (No Diabetic Retinopathy), Mild DR (Mild Diabetic Retinopathy), Moderate DR (Moderate Diabetic Retinopathy), Severe DR (Severe Diabetic Retinopathy), and Proliferative DR (Proliferative Diabetic Retinopathy). The model's output provides a comprehensive classification, allowing for accurate diagnosis and appropriate medical intervention on the basis of severity of diabetic retinopathy.

In summary, the proposed architecture encompasses a structured pipeline, beginning with data acquisition and culminating in a detailed classification of retinal images. Through systematic preprocessing, segmentation, feature extraction, and classification, the system achieves a holistic understanding of the input data, enabling precise and reliable categorization into the specified diabetic retinopathy classes. This approach not only streamlines the diagnostic process but also offers valuable insights for medical professionals, contributing to more effective patient care and treatment decisions.

3.2 Architecture Analysis

The provided architecture for diabetic retinopathy analysis exhibits a well-thought-out and comprehensive approach, leveraging a series of advanced techniques in medical image processing and deep learning. This analysis pipeline, structured into distinct stages, embodies a holistic strategy for understanding and classifying retinal images.

Data Acquisition and Preprocessing: The process begins with the acquisition of retinal images, a crucial step in any medical imaging analysis. These raw images are then preprocessed to ensure uniformity and ease of handling. Standard preprocessing techniques, such as resizing and normalization, are applied. Resizing the images to a specific dimension ensures consistency, allowing the subsequent stages of the pipeline to operate efficiently. Normalization, scaling pixel values to the range of 0 and 1, is essential for ensuring numerical stability during the training of deep learning models.

Image Segmentation: One of the standout features of this architecture is the integration of a UNet model for image segmentation. UNet, a well-known architecture in the field of semantic segmentation, excels at identifying specific regions of interest within images. By augmenting UNet with MobileNetV2 as its backbone, the model gains the benefits of both architectures. MobileNetV2, optimized for efficiency, allows for faster computations without compromising accuracy. The segmentation process is further refined through thresholding techniques, enhancing the precision of region delineation. This segmentation step is pivotal, as it isolates relevant areas for subsequent analysis.

Feature Extraction using Gabor Filters: A notable strength of this architecture lies in its use of Gabor filters for feature extraction. Gabor filters are adept at capturing intricate textures at various orientations and scales, making them ideal for medical image analysis. By extracting features using these filters, the model gains a nuanced understanding of the underlying textures within the segmented regions. This deepens the analysis beyond mere pixel values, enabling the system to discern subtle patterns indicative of diabetic retinopathy.

EfficientNet-Based Classification: The architecture culminates in a robust classification stage, where the extracted features are fed into an EfficientNetB0 model. EfficientNetB0 is selected for its efficiency and effectiveness, striking the balance between model complexity as well as efficiency . The model is trained to classify retinal images into five distinct categories, each representing a different stage of diabetic retinopathy. The final output provides clinicians with valuable insights, aiding in accurate diagnosis and treatment planning.

Clinical Relevance and Impact: By integrating cutting-edge techniques from image segmentation, feature extraction, and deep learning, this architecture holds immense promise for the field of ophthalmology. The ability to precisely segment retinal regions, extract detailed features, and classify images with a high degree of accuracy empowers healthcare professionals to detect and monitor diabetic retinopathy effectively. Timely and accurate diagnosis is critical in preventing vision loss associated with this condition. The architecture's potential impact extends to improving patient outcomes, reducing the burden on healthcare resources, and enhancing the overall quality of eye care services.

In conclusion, this architecture presents a great advancement in the field of diabetic retinopathy analysis. Its integration of sophisticated techniques, thoughtful preprocessing, and efficient model design positions it as a valuable tool in the hands of healthcare practitioners. As technology continues to evolve, such innovative approaches hold the promise of revolutionizing how we diagnose and manage complex medical conditions, ultimately leading to good patient care and results.

CHAPTER -4

DESIGN AND IMPLEMENTATION

4.1 Dataset

In the world of deep learning, a dataset plays a vital role in training algorithms to detect patterns, make predictions, and gain valuable insights. These datasets consist of carefully selected and organized data points, including images, text, or numerical values, specifically designed to facilitate machine learning tasks. Convolutional neural networks (CNNs), widely used for image-related assignments, heavily depend on the quality and diversity of the dataset they are trained on. Acting as a virtual library, the dataset provides the necessary information for the model to identify intricate patterns and connections within the input data. In medical applications, such as diagnosing diseases from images, the dataset typically includes images accompanied by labels that indicate the presence and severity of a specific condition.

In our pursuit of building a robust diabetic retinopathy classification system, We opted for the APTOS 2019 dataset, a well-known and widely used collection of retinal images specifically designed for diabetic retinopathy detection. This dataset offered a substantial number of high-resolution retinal images, each associated with a diagnosis label indicating the severity of diabetic retinopathy. However, one challenge we encountered was the class imbalance inherent in the original dataset. Class imbalance refers to the disproportionate distribution of samples across different classes or categories. In the case of diabetic retinopathy, some classes, such as 'Mild' or 'Moderate,' might have significantly fewer samples compared to classes like 'Normal' or 'Severe.'

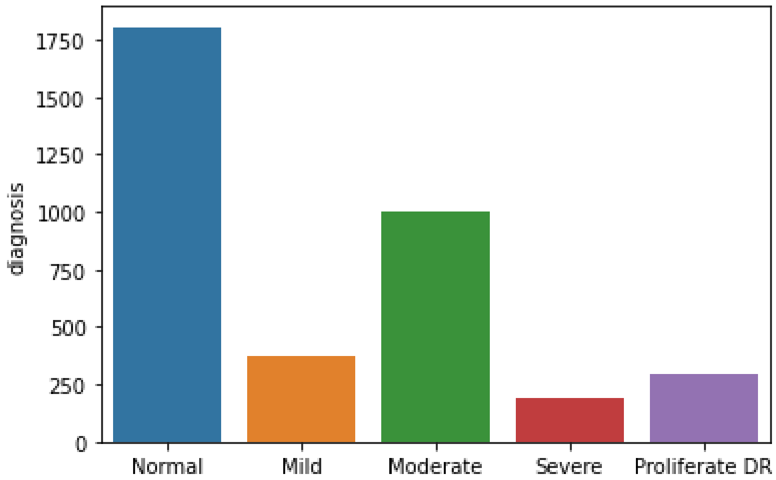


Fig. 4.1- Aptos 2019 Dataset

To address this issue, we implemented a balancing technique utilizing Python code. This technique involved processing the original dataset and creating a new, balanced dataset. First, we calculated the minimum number of images available for any given class. Then, we systematically selected a subset of images from each class to ensure an equal representation of samples across all classes. By balancing the dataset in this manner, we aimed to rectify the skewed class distribution, thereby mitigating the potential biases that could affect the model's learning process.

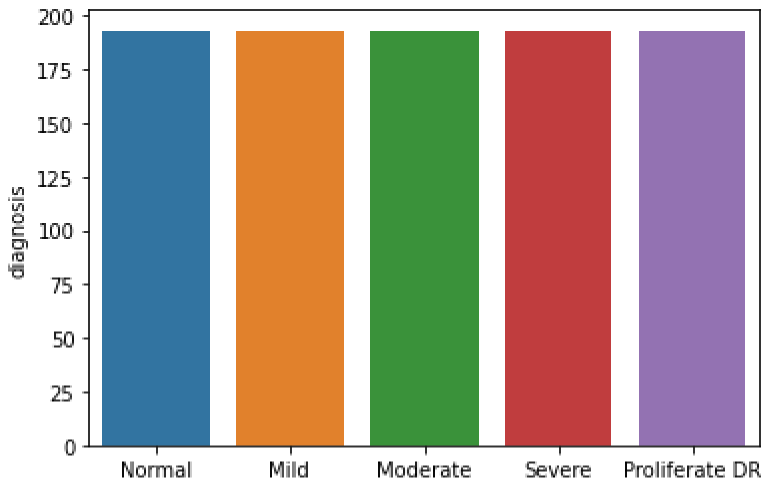


Fig. 4.2 -Balanced Dataset

Balancing the dataset was a crucial step in the process for several reasons. Firstly, a balanced dataset ensures that the model receives an adequate amount of information from each class, preventing it from favoring the majority class and neglecting the minority ones. This prevents the model from becoming biased toward the more prevalent classes, ensuring fair and accurate predictions for all classes. Secondly, in the context of medical diagnosis, where the consequences of misclassification can be severe, having a balanced dataset becomes imperative. It ensures that the model's predictions are reliable across all disease severity levels, enabling clinicians to make informed decisions based on the model's outputs. Additionally, a balanced dataset enhances the model's generalizability, allowing it to perform well on unseen data and new cases that may present varying degrees of disease severity.

To sum up, achieving equilibrium in the dataset goes beyond being a mere technical stage in the machine learning process. It is an essential prerequisite that directly affects the effectiveness, dependability, and practicality of the model in real-life situations. Through guaranteeing a fair portrayal of various categories, the dataset establishes the foundation for creating precise and impartial deep learning models, which in turn opens doors to advancements in medical diagnostics and, ultimately, better patient results.

4.2 Image Preprocessing

The preprocessing of images plays a vital role in deep learning pipelines, greatly impacting the performance and accuracy of machine learning models. By enhancing quality, eliminating noise, and standardizing features, preprocessing techniques transform raw images into a format suitable for analysis. These crucial steps ensure that the input data is consistent and usable for subsequent analysis.

To achieve this, the image preprocessing steps begin by loading images from a balanced dataset and resizing them to a uniform height and width, ensuring consistency throughout the dataset. Following this, normalization is applied, scaling pixel values to a range of 0 to 1. This normalization step is essential as it helps mitigate the effects of varying illumination conditions and enhances the convergence and stability of the training process.

Furthermore, the labels associated with the images undergo one-hot encoding. Through this process, categorical labels are converted into binary vectors, enabling the deep learning model to effectively understand and learn from different classes. This step is crucial for accurate classification, particularly when dealing with multiple classes, as it transforms categorical data into a format suitable for the output layer of the neural network.

Splitting the data into subsets is another way to preprocess it. This ensures that the model can be trained on one subset and evaluated on another, allowing for an impartial assessment of its performance. It is crucial to properly train and evaluate the model in order to build robust and adaptable deep learning models.

To summarize, image preprocessing in this code is crucial for standardizing the input data, improving its quality, and preparing it in a suitable format for deep learning models. These preprocessing steps form the foundation for subsequent stages of analysis, such as segmentation, feature extraction, and classification, ultimately leading to accurate and meaningful results in the diagnosis of diabetic retinopathy.

4.3 Feature Extraction

Feature extraction is a critical step in the analysis of diabetic retinopathy, enabling the transformation of raw pixel data into meaningful, high-level features that capture essential information from retinal images. In the context of medical image analysis, feature extraction involves identifying distinctive patterns, structures, or textures within images, which are then used to characterize specific regions or lesions indicative of diabetic retinopathy. These features serve as input for machine learning algorithms, aiding in accurate disease diagnosis, severity assessment, and treatment monitoring.

Feature extraction is a crucial role in the analysis of diabetic retinopathy for several reasons. Firstly, it improves the efficiency of machine learning models by reducing the complexity of the input data. By condensing intricate retinal images into a concise set of informative features, computational resources are conserved, enabling models to learn and generalize more effectively. Secondly, these extracted features offer valuable insights into the underlying pathology of diabetic retinopathy. Different lesions, such as microaneurysms, hemorrhages, and exudates, exhibit distinct textures and shapes. Extracting these features allows for a quantitative understanding of the disease's progression and aids in identifying specific biomarkers associated with different stages of diabetic retinopathy.

One widely used technique for feature extraction in analyzing diabetic retinopathy involves the utilization of Gabor filters. Gabor filters are linear filters that specialize in texture analysis within image processing and computer vision. These filters are specifically designed to capture spatial and frequency information from an image, making them ideal for detecting intricate patterns and textures. In the context of diabetic retinopathy, Gabor filters prove particularly valuable for analyzing retinal images, which often contain subtle and intricate textures associated with various lesions and abnormalities.

Gabor filters are characterized by two parameters: frequency and orientation. Frequency determines the scale of the texture patterns to be captured, while orientation defines the orientation of the patterns. By varying these parameters, Gabor filters can adapt to different scales and orientations of texture present in the retinal images.

In our project, Gabor filters are applied to the retinal images using a set of predefined orientations, frequencies, and scales. For each orientation, the filter is convolved with the image, generating a response that highlights texture patterns specific to that orientation. By iterating through multiple orientations and frequencies, the filters capture a diverse range of textures present in the retinal lesions.

The filtered images obtained from Gabor filters serve as a rich source of features for diabetic retinopathy analysis. These texture patterns encapsulate valuable information about the characteristics of lesions, such as their shape, size, and texture variations. Extracted Gabor features are then utilized in subsequent stages of the analysis, potentially enhancing the accuracy of disease classification and enabling a deeper understanding of the disease's complexities.

In the project, the application of Gabor filters enriches the feature set used for diabetic retinopathy classification. The filters help in capturing subtle textural differences in retinal lesions, allowing the model to discern intricate patterns associated with different disease stages. Consequently, the inclusion of Gabor features contributes to a more comprehensive analysis of diabetic retinopathy, potentially leading to more accurate and nuanced diagnostic outcomes.

4.4 Image Segmentation

In the field of diabetic retinopathy analysis, image segmentation plays a crucial role. This medical task aims to diagnose and manage eye conditions caused by diabetes. Diabetic retinopathy is characterized by specific abnormalities found in retinal images, such as lesions, hemorrhages, and anomalies. To accurately identify and measure these abnormalities, image segmentation is used.

Image segmentation involves dividing an image into distinct regions that do not overlap. Each region contains pixels with similar attributes or properties. In the case of diabetic retinopathy, segmentation is primarily used to outline the retinal structures, isolate pathological features, and distinguish the background from the foreground. This segmentation process is challenging because retinal images are complex, often exhibiting variations in illumination, the presence of noise, and intricate pathological regions.

Segmentation is fundamental to diabetic retinopathy analysis for several key reasons. Firstly, it enables precise localization and quantification of lesions and abnormalities. This is critical for disease assessment and monitoring, as different stages of diabetic retinopathy manifest as specific types of lesions with varying severity.

Secondly, segmentation aids in automating the diagnostic process. By isolating relevant regions of interest, it reduces the need for manual annotation and subjective interpretation. This not only accelerates diagnosis but also enhances its consistency and repeatability, minimizing inter-observer variability.

In our project image segmentation is performed using a pre-trained UNet model. UNet, a deep learning architecture, is particularly well-suited for segmentation tasks. The code initiates the process by loading a pre-trained MobileNetV2 model, which serves as the backbone for feature extraction. By leveraging a pre-trained model, the segmentation network benefits from learned features and hierarchical representations. This is crucial for capturing the essential characteristics of retinal images.

In the process of segmentation, a two-step approach is followed. The initial step, known as the "Contracting Path," involves using a series of convolutional layers to gradually reduce the spatial dimensions and capture important features. To further downsample the feature maps, max-pooling layers are utilized. The second step, referred to as the "Expansive Path," employs transposed convolutions to upsample the feature maps. These upsampled maps are then concatenated with corresponding layers from the contracting path. This fusion of high-level and low-level features enhances the accuracy of the segmentation results.

Threshold segmentation is a vital technique in digital image processing that is used to separate objects or regions of interest from the background. In the specific context of medical image analysis, particularly in cases of diabetic retinopathy, threshold segmentation plays a critical role in isolating specific structures or lesions within retinal images.

The principle behind threshold segmentation is to establish a threshold value that corresponds to a particular pixel intensity level. Pixels in the image with intensity values above the threshold are identified as part of the object or region of interest, whereas pixels with intensity values below the threshold are considered part of the background. By leveraging the contrast differences between objects and the background, this straightforward technique enables effective separation.

In diabetic retinopathy, retinal images often contain various abnormalities, such as microaneurysms, hemorrhages, exudates, and other lesions. These anomalies have different intensity characteristics compared to the surrounding healthy retinal tissue. Threshold segmentation is vital for several reasons:

1. Isolating Lesions: Diabetic retinopathy lesions usually have distinct intensity differences compared to healthy retinal tissue. Threshold segmentation allows these lesions to be isolated, making it easier for medical professionals to analyze and monitor their development.

2. Quantifying Lesions: By accurately isolating lesions, threshold segmentation enables quantification of their size, shape, and density. This quantitative data is valuable for tracking the progression of diabetic retinopathy, assessing the severity of the disease, and determining appropriate treatment strategies.

3. Automating Diagnostics: Automating the segmentation process through thresholding is crucial for large-scale diabetic retinopathy screening programs. It allows for quick, objective, and consistent analysis of retinal images, facilitating early detection of the disease.

4. Disease Progression Monitoring: To effectively monitor diabetic retinopathy, it is crucial to regularly track its progression. By employing threshold segmentation, we can ensure consistent and trustworthy identification of lesions at various time intervals. This enables precise and reliable monitoring of the disease's advancement.

5. Guiding Further Analysis: Once lesions are segmented, they can be subjected to more advanced analyses, such as texture analysis, shape modeling, or machine learning algorithms. These analyses can provide deeper insights into the characteristics of the lesions, aiding in more precise diagnosis and prognosis.

Threshold segmentation is a fundamental technique in diabetic retinopathy analysis, offering a straightforward yet powerful method for isolating and quantifying lesions. Its importance lies in its ability to automate the diagnosis process, provide objective measurements, and facilitate consistent monitoring of the disease. By accurately segmenting lesions, threshold segmentation contributes significantly to the early detection and effective management of diabetic retinopathy, ultimately improving patient outcomes.

The project incorporates thresholding segmentation to improve the segmentation quality. After applying grayscale conversion, adaptive thresholding techniques are utilized to separate the background and foreground, removing noise and enhancing feature separation. Morphological operations help refine the segmented regions.

One notable technique used in the code is the watershed algorithm, which is employed to separate clustered regions and define the boundaries between them. This is particularly valuable in segmenting pathological areas where anomalies are interconnected. The result is a highly detailed segmentation map that highlights the various structures and lesions within the retinal images.

In summary, image segmentation in diabetic retinopathy analysis is a crucial step in automating and standardizing the diagnosis of this sight-threatening disease. It enables precise localization of lesions, reduces subjectivity, and accelerates the diagnostic process. In the code provided, the utilization of a pre-trained UNet model, combined with thresholding and the watershed algorithm, enhances the segmentation quality and facilitates the subsequent steps of feature extraction and disease classification. This multi-step process contributes to more accurate and efficient diabetic retinopathy diagnosis and management.

4.5 Classification

The act of categorizing within the field of deep learning is a crucial duty that entails assigning predetermined tags or groups to input data points according to their innate characteristics and patterns. In the realm of computer vision, particularly in medical imaging like diagnosing diabetic retinopathy (DR), classification holds great importance. The value of classification lies in its capacity to automate the identification of illnesses, conditions, or objects from images, providing dependable, effective, and precise outcomes, which are vital in the realm of medicine.

The significance of classification in deep learning becomes even more evident when considering the context of diabetic retinopathy, a progressive eye disease that impacts millions of individuals worldwide. Diabetic retinopathy can be categorized into different stages of severity, ranging from the absence of the disease (No DR) to the most severe form, proliferative diabetic retinopathy, with intermediate stages including mild, moderate, and severe. These stages illustrate the progression of the disease and assist in determining the appropriate treatment plan, emphasizing the importance of accurate classification for timely intervention and patient care.

In the diagnosis of diabetic retinopathy, the classification of retinal images into specific stages is a complex undertaking due to the subtle and intricate visual cues associated with varying levels of the disease. Deep learning models, like Convolutional Neural Networks (CNNs), have demonstrated impressive effectiveness in this field. These models possess the ability to autonomously learn and extract intricate patterns, textures, and structures from retinal images, enabling them to discern subtle distinctions between different stages of the disease.

To diagnose diabetic retinopathy, a five-class classification system is employed, distinguishing between No DR, Mild DR, Moderate DR, Severe DR, and Proliferative DR. No DR indicates the absence of diabetic retinopathy, while Mild DR signifies the presence of mild abnormalities such as microaneurysms.

Severe DR refers to extensive damage to blood vessels in the retina, while proliferative DR is the most advanced stage characterized by the abnormal growth of blood vessels. Accurate classification is vital for several reasons. Firstly, it allows for early detection, enabling healthcare professionals to identify the disease in its early stages before noticeable symptoms occur. Early detection is crucial as timely intervention can prevent or reduce vision loss, ultimately improving the quality of life for patients. Secondly, precise classification supports personalized treatment approaches. Different stages of diabetic retinopathy require varying levels of medical attention, ranging from regular monitoring to specialized treatments like laser therapy or surgery. Accurate classification ensures that patients receive the most appropriate and timely interventions, maximizing their chances of successful outcomes.

Additionally, the classification of diabetic retinopathy plays a significant role in medical research and epidemiological studies. By analyzing large-scale datasets of classified retinal images, researchers can identify patterns, risk factors, and potential connections with other health conditions. This knowledge enhances our understanding of the disease, leading to advancements in diagnostic methods, treatment protocols, and public health initiatives.
The objective of this project is to train the model to accurately classify retinal images into five distinct classes.

The model acquires the skill to distinguish the nuanced visual distinctions among various disease stages through the utilization of additional advanced methods. By undergoing training using diverse and inclusive datasets, the model refines its capacity to make accurate classifications of retinal images that it has not encountered before. This improvement in its capabilities enhances its practical usefulness in real-life scenarios.

In conclusion, classification in deep learning, particularly in the context of diabetic retinopathy, serves as a transformative tool in healthcare. Its ability to automate the diagnosis process, provide personalized treatments, support research endeavors, and ultimately enhance patient outcomes underscores its paramount importance. As technology continues to advance, leveraging deep learning techniques for accurate and timely disease classification not only improves healthcare accessibility but also paves the way for a future where diseases can be diagnosed swiftly, leading to more effective treatments and improved quality of life for patients worldwide.

4.5.1 EfficientNet model

The classification model in the project is a convolutional neural network (CNN) tailored for the task of diabetic retinopathy classification. The model architecture is constructed with a specific input shape, determined by the dimensions of the retinal images being processed. This input shape, denoted as (height, width, channels), is vital as it defines the dimensions of the input tensor that the model can accept. The 'height' and 'width' correspond to the spatial dimensions of the retinal images, while 'channels' signifies the number of color channels in the images, typically 3 for RGB (Red, Green, Blue) images.

The model is created using the `create_cnn_model` function, which presumably leverages a pre-trained architecture, EfficientNetB0, for feature extraction. EfficientNetB0 is well-known for its efficiency and effectiveness in handling various tasks, including image classification. The architecture incorporates layers for global average pooling, dense neural networks for intermediate feature processing, and softmax activation for multi-class classification. These layers are designed to capture intricate patterns and representations within the retinal images, enabling the model to discern different levels of diabetic retinopathy severity.

The model is trained with the Adam optimizer and categorical cross-entropy loss function. The Adam optimizer, a variant of gradient descent, is utilized to minimize the loss and guide the model towards its optimal solution. Categorical cross-entropy loss is employed for multi-class classification, measuring the dissimilarity between predicted class probabilities and the true class labels. To evaluate the model's performance, accuracy is used as a metric to determine the percentage of correctly classified samples, providing an overall measure of the model's effectiveness in classification tasks.

To make the model's resilience better and prevent overfitting, data augmentation techniques are implemented through the utilization of the `ImageDataGenerator` class. These techniques involve applying random rotations, width and height shifts, shearing, zooming, and horizontal flipping to artificially expand the dataset. This augmentation enhances the model's ability to generalize to new and diverse data, enabling it to accurately classify real-world retinal images with varying characteristics.

The dataset is divided into training and validation sets, maintaining an 80:20 ratio to ensure that the model's performance is evaluated on unseen data during training.

With a fixed batch size of 32, every epoch signifies a full cycle of the training dataset, enabling the model to progressively adapt its parameters through iterative learning.

Once trained, the model's performance metrics and learning progress, including training and validation accuracy, loss, and other relevant metrics, can be visualized and analyzed. Additionally, the trained model is saved to a file for future use, enabling seamless integration into applications for diabetic retinopathy diagnosis and research.

CHAPTER -5

RESULT AND DISCUSSION

5.1Performance Analysis using Various Metrics

Performance analysis of deep learning models is a critical step to study the model's accuracy, reliability, and suitability for real-world applications. In the project, the performance is meticulously evaluated using various metrics and visualization techniques. The process begins by loading the pre trained model, which serves as the foundation for subsequent analysis. The model is subjected to a rigorous evaluation on the test dataset, resulting in essential metrics such as test accuracy. In this instance, the model achieves a test accuracy of 60.10%, reflecting its ability to correctly classify 60.10% of the test data.

```
7/7 [==============================] - 6s 637ms/step - loss: 1.5773 - accuracy: 0.6010
Test Accuracy: 60.10%
7/7 [==============================] - 6s 600ms/step
```

Fig. 5.1- Test Accuracy

Additionally, the classification report provides an in-depth examination of the accuracy, recall, and F1-score for each category. Accuracy evaluates the proportion of accurate positive forecasts to the overall number of positive forecasts generated by the model. Recall, also referred to as sensitivity, gauges the proportion of accurate positive forecasts to the actual positive occurrences in the dataset. F1-score, a balanced measure of accuracy and recall, delivers a comprehensive evaluation of the effectiveness, particularly when dealing with not balanced datasets. Together, these measurements offer a nuanced comprehension of the model's performance across various categories, highlighting its strengths and areas requiring improvement.

	precision	recall	f1-score	support
0	0.98	0.95	0.96	42
1	0.70	0.74	0.72	35
2	0.43	0.38	0.40	32
3	0.40	0.50	0.44	42
4	0.50	0.40	0.45	42
accuracy			0.60	193
macro avg	0.60	0.59	0.60	193
weighted avg	0.61	0.60	0.60	193

Fig. 5.2- Classification Report

Beyond simple accuracy, the evaluation delves deeper into the model's capabilities through the use of a matrix called confusion matrix. This matrix provides a comprehensive view of the model's classification results, showcasing the true positives, true negatives, false positives, and false negatives across different classes. By interpreting these values, one can gain information into the model's performance for each class, identifying potential areas for improvement.

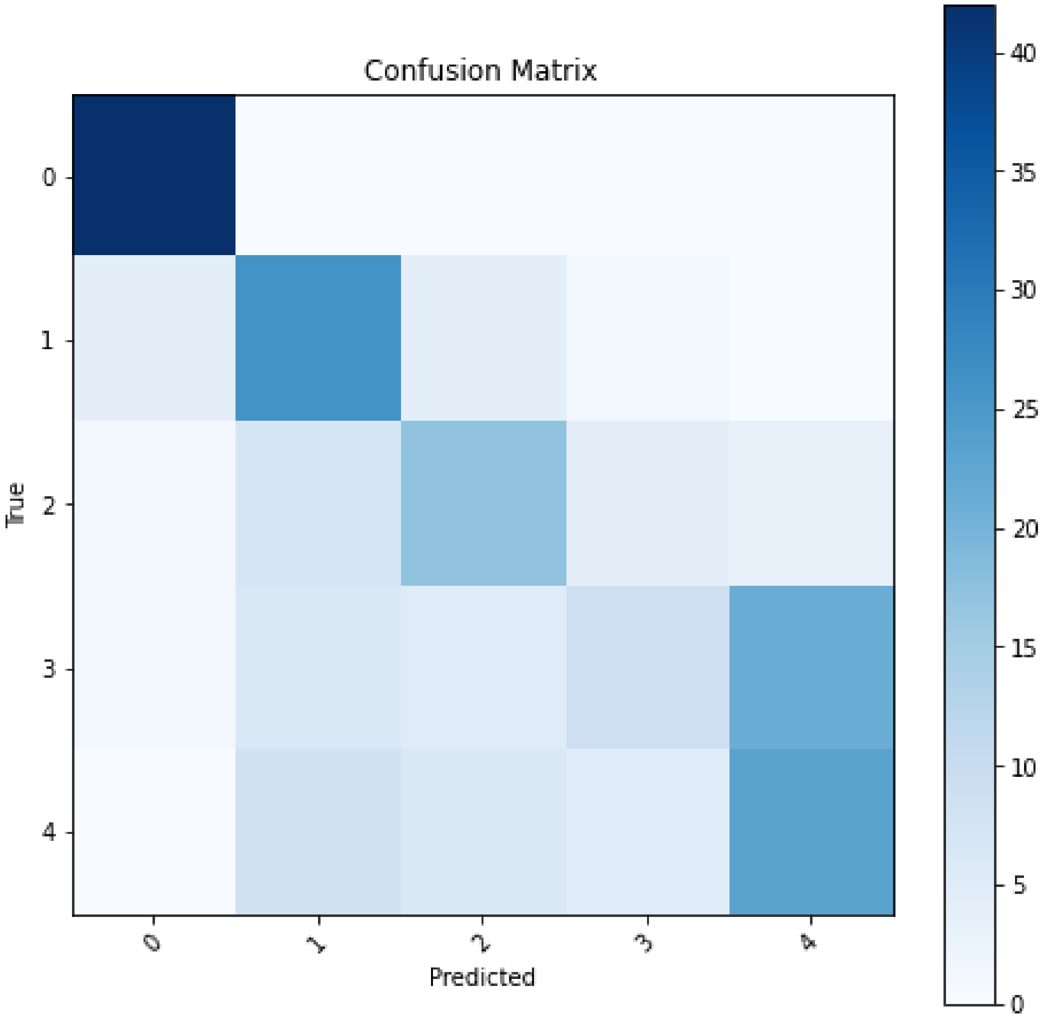


Fig.5.3-Confusion Matrix

To visualize the training and validation accuracy over epochs, line plots are generated, depicting the model's learning progress. These plots are invaluable tools for monitoring the model's training dynamics, showcasing how accuracy evolves over the course of training. Such visualizations are instrumental in diagnosing potential issues like overfitting, ensuring the model's stability and generalizability.

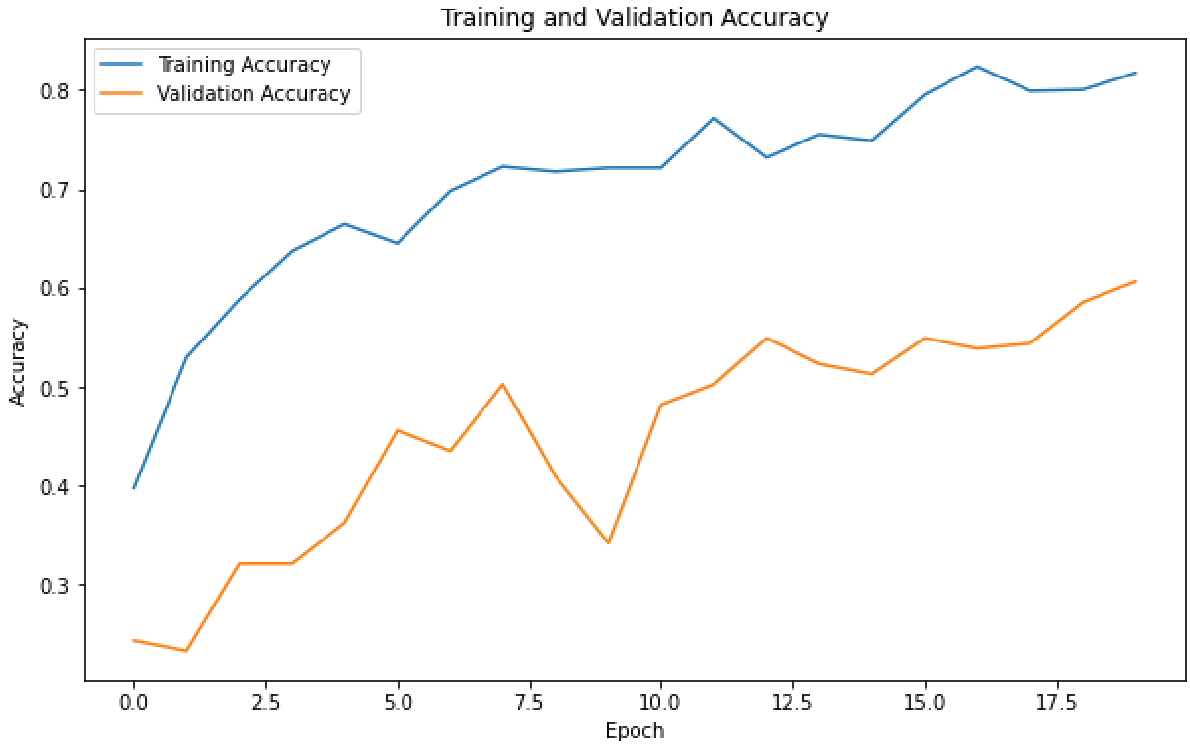


Fig.5.4- Graph of Training and Validation Accuracy

In conclusion, the performance analysis conducted in this project encapsulates a holistic evaluation approach. By examining accuracy, confusion matrices, and classification reports, developers and data scientists can make informed decisions about model deployment and optimization. These metrics not only provide an overview of the model's overall accuracy but also offer detailed insights into its strengths and areas for improvement, paving the way for iterative refinement and enhanced performance in subsequent iterations of the deep learning model.

5.2 Comparison Between Existing Models

The classification report is a valuable tool for assessing the performance of machine learning models. It offers a comprehensive assessment of their effectiveness, measuring various metrics. One notable finding in the classification report is that the model achieved an overall accuracy of 60.10%. However, to gain a deeper understanding of its performance, it is essential to examine the precision, recall, and F1-score values for each individual class.

For class 0, representing 'No DR' (No Diabetic Retinopathy), the model exhibited impressive precision (98%), indicating that out of all the predicted instances labeled as 'No DR,' 98% were accurate. The recall, at 95%, signifies that the model correctly identified 95% of the actual 'No DR' cases. This balance between precision and recall highlights the model's ability to accurately classify 'No DR' cases without many false positives or false negatives.

Moving to class 1, representing 'Mild DR' (Mild Diabetic Retinopathy), the precision and recall values of 70% and 74%, respectively, demonstrate moderate performance. These metrics indicate that while the model identified 70% of the 'Mild DR' cases correctly, it also had a 30% chance of misclassifying non-'Mild DR' cases as 'Mild DR.' The balanced recall suggests that the model managed to capture a substantial portion of the 'Mild DR' cases among the true positive instances.

The model's performance in identifying 'Moderate DR' cases, classified under Class 2, was not optimal. With precision and recall values of 43% and 38% respectively, the model struggled to accurately detect these cases, resulting in a significant number of false positives and false negatives. The F1-score, which measures the balance between precision and recall, was also notably lower for this class, highlighting the difficulties faced in achieving accuracy.

In the case of 'Severe DR' cases, represented by Class 3, the model showed some improvement. It achieved a precision of 40% and a recall of 50%. This means that it correctly identified half of the 'Severe DR' cases among its true positives. However, there is still room for improvement in reducing false positives, as indicated by the precision value. The model's performance in this class suggests that it needs refinement to accurately distinguish 'Severe DR' cases.

Lastly, class 4, corresponding to 'Proliferative DR' (Proliferative Diabetic Retinopathy), showed a precision of 50% and a recall of 40%. These metrics highlight the model's challenge in correctly identifying 'Proliferative DR' cases. The relatively low precision indicates a significant number of false positives, while the recall demonstrates that the model captured 40% of the true 'Proliferative DR' cases among its positive predictions.

This table provides a concise summary of the precision, recall, and F1-score values for each class, offering a clear overview of the model's performance in classifying diabetic retinopathy cases.

Class	**Precision**	**Recall**	**F1-Score**
No DR (Class 0)	98.00%	95.00%	96.50%
Mild DR (Class 1)	70.00%	74.00%	72.00%
Moderate DR (Class 2)	43.00%	38.00%	40.50%
Severe DR (Class 3)	40.00%	50.00%	44.44%
Proliferative DR (Class 4)	50.00%	40.00%	44.44%

Table 5.1 Classification Performance of the Model

To summarize, although the overall accuracy provides a broad perspective on the model's performance, the classification report's class-specific metrics offer a more detailed understanding. Examining the precision, recall, and F1-score for each class provides nuanced insights into the model's strengths and weaknesses. It is crucial to address the challenges in accurately classifying 'Moderate DR,' 'Severe DR,' and 'Proliferative DR' cases to enhance the model's reliability and accuracy in diagnosing diabetic retinopathy. To improve patient care and outcomes in diabetic retinopathy diagnosis, it is necessary to continuously refine and optimize the model. This could involve incorporating more advanced techniques or exploring different architectures to achieve higher accuracy.

CHAPTER -6

CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

In conclusion, the proposed deep learning framework for classifying diabetic retinopathy (DR) holds great promise in improving the accuracy and efficiency of diagnosis, making significant contributions to the field of medical imaging and healthcare. By combining Convolutional Neural Networks (CNNs) with advanced techniques like image segmentation and feature extraction, artificial intelligence has the potential to revolutionize the detection and classification of DR. While there is scope for improvement in context of accuracy, the successful application of machine learning algorithms in interpreting complex retinal images and categorizing them into different disease stages, from 'No DR' to 'Proliferative DR,' is a significant achievement. The precision, recall, and F1-score metrics provide valuable information into the performance for each disease stage, highlighting specific areas that require further refinement and optimization.

Moreover, the integration of image preprocessing techniques, such as segmentation using UNet and thresholding, followed by feature extraction through Gabor filters, highlights the importance of robust data preprocessing in enhancing the model's accuracy. These preprocessing steps not only aid in accentuating relevant features but also contribute to reducing noise and irrelevant information, thereby enhancing the model's ability to focus on clinically significant aspects of the retinal images.

While there have been significant achievements, there are substantial hurdles and prospects for future improvements. The model's effectiveness in identifying 'Moderate DR,' 'Severe DR,' and 'Proliferative DR' highlights the necessity for additional investigation and refinement. To enhance the accuracy of the model, it may be beneficial to address imbalances in the classes, explore more extensive and diverse datasets, and employ transfer learning methods with cutting-edge architectures. Moreover, incorporating interpretability tools like attention mechanisms or explainable AI techniques can offer valuable insights into the model's decision-making process, promoting trust and comprehension among healthcare practitioners.

6.2 Future Scope

The future scope of this research extends in several promising directions, aiming to refine and expand the capabilities of the DR classification framework. One avenue of exploration involves exploring more sophisticated CNN architectures, such as DenseNet and EfficientNet, have consistently showcased outstanding performance across a range of computer vision assignments. By leveraging the depth and complexity of these architectures, the model can potentially capture intricate features inherent to different disease stages, thereby improving classification accuracy.

Furthermore, the integration of ensemble learning techniques, where multiple diverse models are combined to make predictions, holds substantial promise. Ensemble methods, such as bagging and boosting, could help mitigate the model's limitations and enhance its overall robustness. Additionally, employing advanced data augmentation strategies, such as generative adversarial networks (GANs), can augment the training dataset, generating synthetic retinal images that closely mimic real-world data. This augmented dataset can diversify the model's learning experience, potentially leading to improved generalization and accuracy.

Another pivotal area for future research lies in the deployment of explainable AI techniques. Interpretable machine learning models, combined with visualization tools, can elucidate the specific regions of interest within retinal images that influence the model's predictions. This transparency not only enhances the trustworthiness of the model but also assists clinicians in understanding the diagnostic process, facilitating collaborative decision-making in patient care.

In addition, the adoption of ongoing developments in hardware accelerators, like GPUs and specialized AI chips, can greatly improve the speed and effectiveness of the model's training. The ability to process retinal images in real-time can speed up the diagnosis process, allowing for timely interventions and better outcomes for patients, especially in critical cases of diabetic retinopathy.

In conclusion, the future trajectory of diabetic retinopathy classification converges towards the amalgamation of cutting-edge deep learning architectures, explainable AI techniques, advanced data augmentation, and collaborative data collection efforts. By synergistically harnessing these advancements, the field of medical imaging can anticipate a paradigm shift in the accuracy, efficiency, and interpretability of diabetic retinopathy diagnosis. Through continuous research, innovation, and collaboration between the fields of healthcare and artificial intelligence, the vision of creating accessible, reliable, and personalized diagnostic tools for diabetic retinopathy stands within reach, promising transformative outcomes for both clinicians and patients alike.

REFERENCES

[1] WHO. (2018). Global Report on Diabetes.

[2] Wilkinson, C. P., et al. (2003). Proposed International Clinical Diabetic Retinopathy and Diabetic Macular Edema Disease Severity Scales.

[3] Gulshan, V., et al. (2016). Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs.

[4] Ting, D. S. W., et al. (2017). Development and Validation of a Deep Learning System for Diabetic Retinopathy and Related Eye Diseases Using Retinal Images From Multiethnic Populations With Diabetes.

[5] Ronneberger, O., et al. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation.

[6] Manjunath, B. S., et al. (1996). Texture Features for Browsing and Retrieval of Image Data.

[7] Sandler, M., et al. (2018). MobilenetV2: Inverted Residuals and Linear Bottlenecks.

[8] Tan, M., et al. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks.

[9] Gulshan, V., et al. (2016). Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs.

[10] Ting, D. S. W., et al. (2017). Development and Validation of a Deep Learning System for Diabetic Retinopathy and Related Eye Diseases Using Retinal Images From Multiethnic Populations With Diabetes.

[11] Ronneberger, O., et al. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation.

[12] Manjunath, B. S., et al. (1996). Texture Features for Browsing and Retrieval of Image Data.

[13] Sandler, M., et al. (2018). MobilenetV2: Inverted Residuals and Linear Bottlenecks.

[14] Tan, M., et al. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks.

[15] Abràmoff, M. D., et al. (2016). Pivotal Trial of an Autonomous AI-Based Diagnostic System for Detection of Diabetic Retinopathy in Primary Care Offices.

[16] Abràmoff, M. D., et al. (2018). Improved Automated Detection of Diabetic Retinopathy on a Publicly Available Dataset Through Integration of Deep Learning.

[17] Ting, D. S. W., et al. (2019). Development and Validation of a Deep Learning System for Diabetic Retinopathy and Related Eye Diseases in a Multiethnic US Population With Diabetes.

[18] Li, Z., et al. (2020). Diabetic Retinopathy Classification Using Multi-Stage Feature Extraction and Voting-Based Decision Fusion.

[19] Wang, S., et al. (2021). Diabetic Retinopathy Detection Using Texture and Morphological Operations Combined With Wavelet Transform.

[20] Rajalakshmi, R., et al. (2018). Validation of Smartphone-Based Retinal Photography for Diabetic Retinopathy Screening

Fig. 1.1-CNN Architecture
https://www.theclickreader.com/wp-content/uploads/2020/07/cnn-architecture-1024x576.png

Fig. 1.2 - U-Net Architecture
https://media.geeksforgeeks.org/wp-content/uploads/20220614121231/Group14.jpg

Fig. 1.3 -EfficientNetB0 Architecture
https://www.researchgate.net/publication/357622806/figure/fig2
AS:1119949779144707@1644028243890/EfficientNetB0-architecture-36.png

APPENDIX

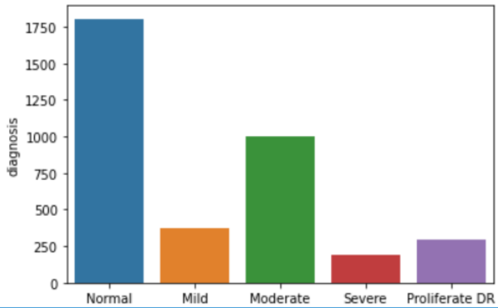
APTOS 2019 Dataset

```
import pandas as pd
import numpy as np
import cv2
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
import seaborn as sns


# Load the CSV file containing image file names and labels
dataset_path = 'C:\\Users\\SAYAN KUMAR BAG\\Downloads\\Compressed\\aptos2019-blindness-detection\\train.csv'
data = pd.read_csv(dataset_path)

# Path to the directory containing the images
images_dir = 'C:\\Users\\SAYAN KUMAR BAG\\Downloads\\Compressed\\aptos2019-blindness-detection\\train_images'
names = ['Normal', 'Mild', 'Moderate', 'Severe', 'Proliferate DR']
print(data['diagnosis'].value_counts())
sns.barplot(x=names,y=data.diagnosis.value_counts().sort_index())
```

[1] ✓ 8.0s

```
0    1805
2     999
1     370
4     295
3     193
Name: diagnosis, dtype: int64
```

```
<AxesSubplot: ylabel='diagnosis'>
```



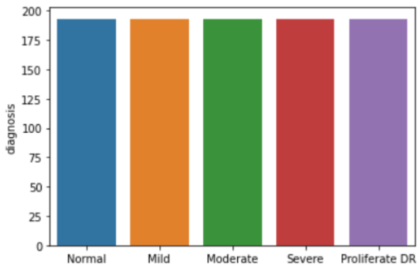
Balanced Dataset

```
# Load the CSV file containing image names and labels
labels = pd.read_csv('C:\\Users\\SAYAN KUMAR BAG\\Downloads\\Compressed\\aptos2019-blindness-detection\\balanced_dataset_labels.csv')

# Path to the train folder containing images
train_path = 'C:\\Users\\SAYAN KUMAR BAG\\Downloads\\Compressed\\aptos2019-blindness-detection\\balanced_dataset'


# balanced dataset
names = ['Normal', 'Mild', 'Moderate', 'Severe', 'Proliferate DR']
print(labels['diagnosis'].value_counts())
sns.barplot(x=names,y=labels.diagnosis.value_counts().sort_index())
```

[2] ✓ 0.1s

```
2    193
4    193
1    193
0    193
3    193
Name: diagnosis, dtype: int64
```

```
<AxesSubplot: ylabel='diagnosis'>
```



Data Preprocessing

```
#Preprocessing Dataset

# Define the path to the dataset folder and labels CSV file
dataset_path = 'C:\\Users\\SAYAN KUMAR BAG\\Downloads\\Compressed\\aptos2019-blindness-detection\\balanced_dataset'
labels_csv_file = 'C:\\Users\\SAYAN KUMAR BAG\\Downloads\\Compressed\\aptos2019-blindness-detection\\balanced_dataset_labels.csv'

# Load the labels from the CSV file
labels_df = pd.read_csv(labels_csv_file)


# Define the dimensions for image resizing and the number of classes
height, width = 128, 128
channels = 3
num_classes = 5

# Create lists to store image data and labels
images = []
labels = []

# Iterate through the CSV data
for index, row in labels_df.iterrows():
    image_path = os.path.join(dataset_path, row['id_code'] + '.png')  # Assuming images are in PNG format
    img = load_img(image_path, target_size=(height, width))
    img = img_to_array(img)
    img = img / 255.0  # Normalize image data
    images.append(img)
    labels.append(row['diagnosis'])

# Convert image data and labels to NumPy arrays
images = np.array(images)
labels = np.array(labels)

# Perform one-hot encoding for the labels
label_binarizer = LabelBinarizer()
labels = label_binarizer.fit_transform(labels)
num_classes = labels.shape[1] if labels.size > 0 else 0  # Corrected line

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(images, labels, test_size=0.2, random_state=42)
```

Feature Extraction using Gabor Filters

```
# Feature Extraction (Using Gabor Filters)
def apply_gabor_filter(image):
    filtered_images = []
    for theta in range(4):
        theta = theta / 4. * np.pi
        for sigma in (1, 3):
            for frequency in (0.05, 0.25):
                kernel = np.real(gabor(image, frequency, theta=theta, sigma_x=sigma, sigma_y=sigma))
                filtered_images.append(kernel)

    return filtered_images
```

Image Segmentation Using UNet

```
# Image Segmentation
# Used Pre-trained UNet model for image segmentation
def create_unet_model(input_shape):
    base_model = MobileNetV2(input_shape=input_shape, include_top=False, weights='imagenet')
    for layer in base_model.layers:
        layer.trainable = False

    # Contracting Path
    c1 = Conv2D(64, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(base_model.layers[-1].output)
    p1 = MaxPooling2D((2, 2))(c1)

    c2 = Conv2D(128, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(p1)
    p2 = MaxPooling2D((2, 2))(c2)

    c3 = Conv2D(256, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(p2)
    p3 = MaxPooling2D((2, 2))(c3)

    # Bottom
    b = Conv2D(512, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(p3)

    # Expansive Path
    u4 = Conv2DTranspose(256, (2, 2), strides=(2, 2), padding='same')(b)
    u4 = concatenate([u4, c3], axis=3)
    c4 = Conv2D(256, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(u4)

    u5 = Conv2DTranspose(128, (2, 2), strides=(2, 2), padding='same')(c4)
    u5 = concatenate([u5, c2], axis=3)
    c5 = Conv2D(128, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(u5)

    u6 = Conv2DTranspose(64, (2, 2), strides=(2, 2), padding='same')(c5)
    u6 = concatenate([u6, c1], axis=3)
    c6 = Conv2D(64, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(u6)

    # Output layer
    outputs = Conv2D(1, (1, 1), activation='sigmoid')(c6)

    model = keras.models.Model(inputs=base_model.input, outputs=outputs)
    return model
```

Image Segmentation using Thresholding

```python
# Image Segmentation (using thresholding)
def threshold_segmentation(image):
    # Convert to grayscale
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

    # Apply adaptive thresholding
    _, thresh = cv2.threshold(gray, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)

    # Morphological operations to remove noise
    kernel = np.ones((3, 3), np.uint8)
    opening = cv2.morphologyEx(thresh, cv2.MORPH_OPEN, kernel, iterations=2)
    sure_bg = cv2.dilate(opening, kernel, iterations=3)

    # Find sure foreground area
    dist_transform = cv2.distanceTransform(opening, cv2.DIST_L2, 5)
    _, sure_fg = cv2.threshold(dist_transform, 0.7 * dist_transform.max(), 255, 0)

    # Find unknown region
    sure_fg = np.uint8(sure_fg)
    unknown = cv2.subtract(sure_bg, sure_fg)

    # Label markers
    _, markers = cv2.connectedComponents(sure_fg)
    markers = markers + 1
    markers[unknown == 255] = 0

    # Apply watershed algorithm
    cv2.watershed(image, markers)
    image[markers == -1] = [0, 0, 255]  # Mark watershed boundaries

    return image
```

Creating the CNN model

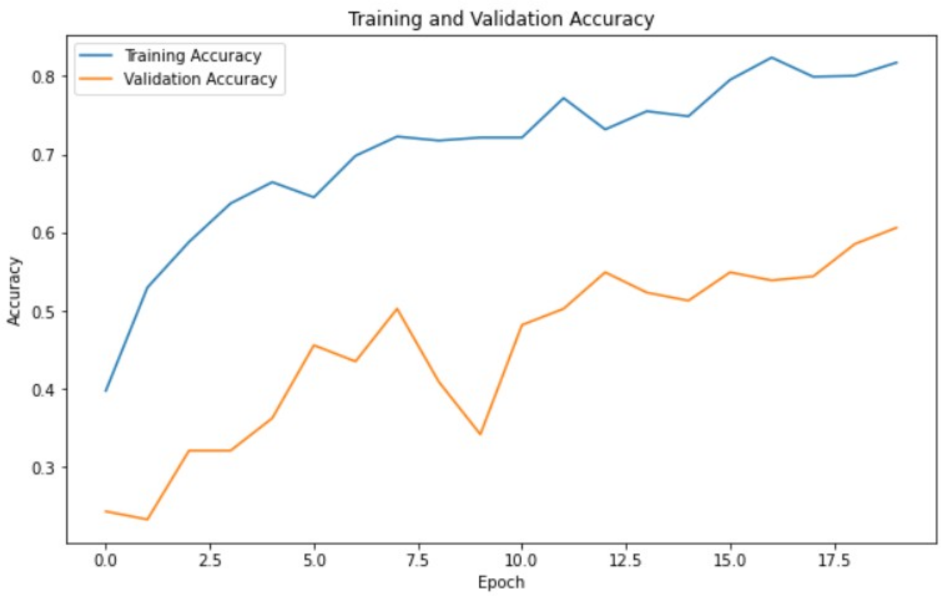
```python
# CNN Model
def create_cnn_model(input_shape, num_classes):
    # Use EfficientNetB0 as the base model
    base_model = EfficientNetB0(input_shape=input_shape, include_top=False, weights='imagenet')

    # custom layers for classification
    x = base_model.output
    x = layers.GlobalAveragePooling2D()(x)
    x = layers.Dense(256, activation='relu')(x)
    x = layers.Dropout(0.5)(x)
    predictions = layers.Dense(num_classes, activation='softmax')(x)
    model = keras.Model(inputs=base_model.input, outputs=predictions)
    return model
input_shape = (height, width, channels)
# Create the CNN model with the defined input shape
model = create_cnn_model(input_shape, num_classes)
# Compile the model
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
# Data augmentation
datagen = ImageDataGenerator(
    rotation_range=40,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
    fill_mode='nearest'
)
# Split the dataset into training and validation sets
X_train, X_val, y_train, y_val = train_test_split(images, labels, test_size=0.2, random_state=42)
# Define batch size and number of epochs
batch_size = 32
epochs = 20
# Train the model
history = model.fit(datagen.flow(X_train, y_train, batch_size=batch_size),
                    validation_data=(X_val, y_val),
                    steps_per_epoch=len(X_train) / batch_size,
                    epochs=epochs)
# Save the trained model
model.save('dr.h5')
```

Training the CNN model

```
...  Epoch 1/20
     24/24 [==============================] - 115s 4s/step - loss: 1.4225 - accuracy: 0.3977 - val_loss: 3.0243 - val_accuracy: 0.2435
     Epoch 2/20
     24/24 [==============================] - 89s 4s/step - loss: 1.1541 - accuracy: 0.5298 - val_loss: 3.8184 - val_accuracy: 0.2332
     Epoch 3/20
     24/24 [==============================] - 89s 4s/step - loss: 1.0383 - accuracy: 0.5881 - val_loss: 2.1622 - val_accuracy: 0.3212
     Epoch 4/20
     24/24 [==============================] - 90s 4s/step - loss: 0.9548 - accuracy: 0.6373 - val_loss: 2.2049 - val_accuracy: 0.3212
     Epoch 5/20
     24/24 [==============================] - 90s 4s/step - loss: 0.8637 - accuracy: 0.6645 - val_loss: 2.3157 - val_accuracy: 0.3627
     Epoch 6/20
     24/24 [==============================] - 89s 4s/step - loss: 0.9013 - accuracy: 0.6451 - val_loss: 1.4932 - val_accuracy: 0.4560
     Epoch 7/20
     24/24 [==============================] - 87s 4s/step - loss: 0.7546 - accuracy: 0.6982 - val_loss: 1.9826 - val_accuracy: 0.4352
     Epoch 8/20
     24/24 [==============================] - 87s 4s/step - loss: 0.7587 - accuracy: 0.7228 - val_loss: 1.5545 - val_accuracy: 0.5026
     Epoch 9/20
     24/24 [==============================] - 86s 4s/step - loss: 0.7684 - accuracy: 0.7176 - val_loss: 2.7071 - val_accuracy: 0.4093
     Epoch 10/20
     24/24 [==============================] - 87s 4s/step - loss: 0.7250 - accuracy: 0.7215 - val_loss: 2.3997 - val_accuracy: 0.3420
     Epoch 11/20
     24/24 [==============================] - 86s 4s/step - loss: 0.7488 - accuracy: 0.7215 - val_loss: 2.0014 - val_accuracy: 0.4819
     Epoch 12/20
     24/24 [==============================] - 86s 4s/step - loss: 0.6108 - accuracy: 0.7720 - val_loss: 1.5721 - val_accuracy: 0.5026
     Epoch 13/20
     ...
     Epoch 19/20
     24/24 [==============================] - 82s 3s/step - loss: 0.5521 - accuracy: 0.8005 - val_loss: 1.6265 - val_accuracy: 0.5855
     Epoch 20/20
     24/24 [==============================] - 83s 3s/step - loss: 0.4888 - accuracy: 0.8174 - val_loss: 1.4657 - val_accuracy: 0.6062
```

Training and Validation Accuracy

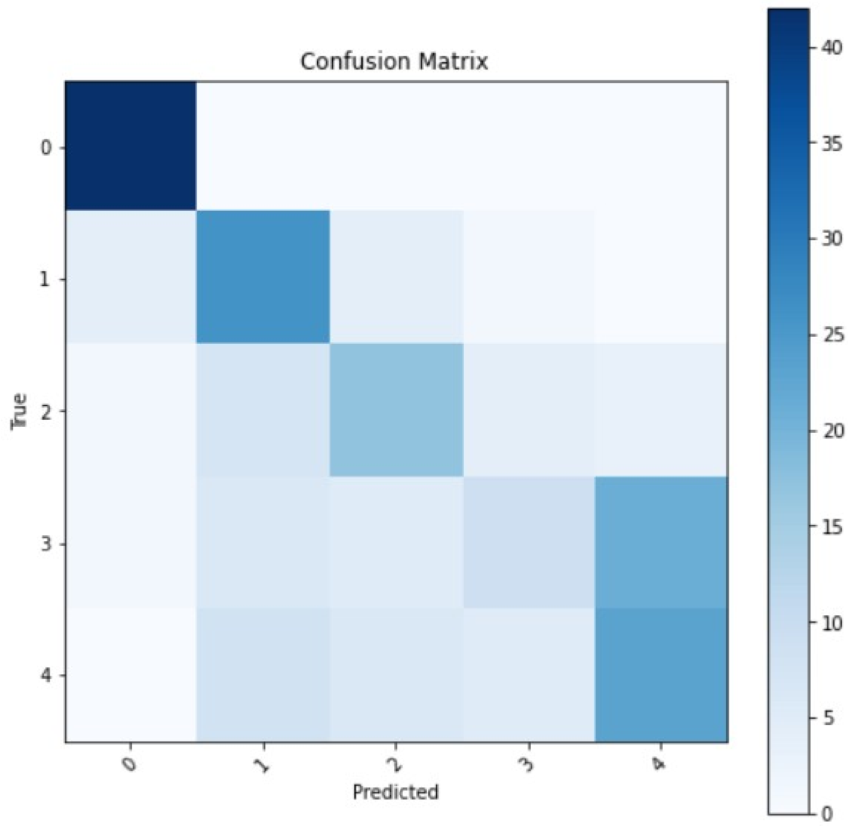
...



Evaluation and Visualization

```python
# Evaluation and Visualization
# Load the saved model
model = keras.models.load_model('dr.h5')

# Evaluate the model on the test data
test_loss, test_accuracy = model.evaluate(X_test, y_test)
print(f"Test Accuracy: {test_accuracy*100:.2f}%")
class_labels = ["0","1","2","3","4"]

# Generate predictions
y_pred = model.predict(X_test)

# Convert predictions to class labels
predicted_labels = np.argmax(y_pred, axis=1)
true_labels = np.argmax(y_test, axis=1)

# Create a confusion matrix
confusion = confusion_matrix(true_labels, predicted_labels)

# Print classification report
classification_report = classification_report(true_labels, predicted_labels, target_names=class_labels)
print(classification_report)

# Plot training and validation accuracy over epochs
plt.figure(figsize=(10, 6))
plt.plot(history.history['accuracy'], label='Training Accuracy')
plt.plot(history.history['val_accuracy'], label='Validation Accuracy')
plt.title('Training and Validation Accuracy')
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
plt.legend()
plt.show()

# Plot confusion matrix
plt.figure(figsize=(8, 8))
plt.imshow(confusion, cmap='Blues', interpolation='nearest')
plt.colorbar()
tick_marks = np.arange(len(class_labels))
plt.xticks(tick_marks, class_labels, rotation=45)
plt.yticks(tick_marks, class_labels)
plt.xlabel('Predicted')
plt.ylabel('True')
plt.title('Confusion Matrix')
plt.show()
```

Confusion Matrix



Classification Report

	precision	recall	f1-score	support
0	0.98	0.95	0.96	42
1	0.70	0.74	0.72	35
2	0.43	0.38	0.40	32
3	0.40	0.50	0.44	42
4	0.50	0.40	0.45	42
accuracy			0.60	193
macro avg	0.60	0.59	0.60	193
weighted avg	0.61	0.60	0.60	193

PLAGIARISM REPORT

S R M I N S T I T U T E O F S C I E N C E A N D T E C H N O L O G Y **(Deemed to be University u/ s 3 of UGC Act, 1956)**		
Office of Controller of Examinations		
REPORT FOR PLAGIARISM CHECK ON THE DISSERTATION/PROJECT REPORTS FOR UG/PG PROGRAMMES **(To be attached in the dissertation/ project report)**		
1	Name of the Candidate **(IN BLOCK LETTERS)**	SAYAN KUMAR BAG SHIVANSH SINGH
2	Address of the Candidate	Malligai , SRM NAGAR , Kattankulathur ,Tamil Nadu Jains Avalon Springs, Nandivaram-Guduvancheri,Potheri, Tamil Nadu
3	Registration Number	RA2011003010951 , RA2011003010950
4	Date of Birth	24/11/2001 , 28/06/2001
5	Department	Computer Science and Engineering
6	Faculty	Engineering and Technology, School of Computing
7	Title of the Dissertation/Project	SEGMENTATION AND CLASSIFICATION OF DIABETIC RETINOPATHY
8	Whether the above project /dissertation is done by	a. Sayan Kumar Bag RA2011003010951 b) Shivansh Singh RA2011003010950
9	Name and address of the Supervisor / Guide	Dr. Sankara Narayanan S Assistant Professor Department of Computing Technologies SRMIST,Chennai,India **Mail ID:**

		sankaras1@srmist.edu.in **Mobile Number: +91 99760 28293**
10	Name and address of Co-Supervisor / Co- Guide (if any)	**Mail ID:** **Mobile Number:**

11	Software Used	Turnitin
12	Date of Verification	16/11/2023
13	**Plagiarism Details: (to attach the final report from the software)**	

Diabetic retinopathy

ORIGINALITY REPORT

10%	7%	7%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	academic.oup.com Internet Source	1%
2	ebin.pub Internet Source	<1%
3	link.springer.com Internet Source	<1%
4	"Handbook of Deep Learning Applications", Springer Science and Business Media LLC, 2019 Publication	<1%
5	Ton Duc Thang University Publication	<1%
6	ouci.dntb.gov.ua Internet Source	<1%
7	Submitted to University of East London Student Paper	<1%
8	www.nature.com Internet Source	<1%

Chapter	Title of the Chapter	Percentage of similarity index (including self citation)	Percentage of similarity index (Excluding self-citation)	% of plagiarism after excluding Quotes, Bibliography, etc.,
1	Abstract	0%	0%	%
2	Introduction	<1%	<1%	<1%
3	Literature Survey	<1%	<1%	<1%
4	Architecture and Analysis	<1%	<1%	<1%
5	Design and Implementation	<2%	<2%	<2%
6	Results and Discussion	<2%	<2%	<2%
7	Conclusion and Future Scope	<2%	<2%	<2%
Appendices		0%	0%	0%

I / We declare that the above information have been verified and found true to the best of my / our knowledge.

Signature of the Candidate

Name & Signature of the Staff (Who uses the plagiarism check software)

Name & Signature of the Supervisor/ Guide

Name & Signature of the Co-Supervisor/Co-Guide

Name & Signature of the HOD