RAPID ANALYSIS OF DIABETIC RETINOPATHY FROM DIGITAL FUNDUSIMAGES USING DEEP CONVOLUTIONAL NEURAL NETWORK

Lakshman Chowdary Moparthy[a], Saaketh Hota*[b], Sai Anurag Komarraj*[c], Shashank Reddy Mallu*[d],Ushasree Dupakuntala*[e]

[a-e] Lakshman Chowdary Moparthy, Saaketh Hota, Sai Anurag Komarraj, Shashank Reddy Mallu, Ushasree Dupakuntala

Computer Science and Engineering

Gokaraju Rangaraju Institute Of Engineering and Technology

E-mail: ((lakshman.moparthy@gmail.com, saaketh.hota@gmail.com, saianurag708@gmail.com, mallushashankreddy@gmail.com, dupakuntlausha@gmail.com))

Abstract: Diabetic Retinopathy (DR) has recently become a major concern for people with diabetes, it is an eye condition that occurred due to severely damaged blood vessels in the retina, leading to visual impairment. According to the projections from the World Health Organization, DR would affect 235 million people by 2040. This research paper presents a convolutional neural network (CNN) model using DenseNet121 architecture, making it more efficient to train for the task of image classification and detecting Diabetic Retinopathy using color fundus images. In this regard, we proposed a model to automate the detection of anomalies in retinal images using cost-effective image processing techniques and also to address the challenges of real-world data. The dataset used for this paper is EYEPACS, consisting of 35,128 color fundus images, which are widely used for training and testing models for detecting Diabetic Retinopathy. The metrics that are used for measuring the performance of our model are accuracy, precision, and F1-Score.

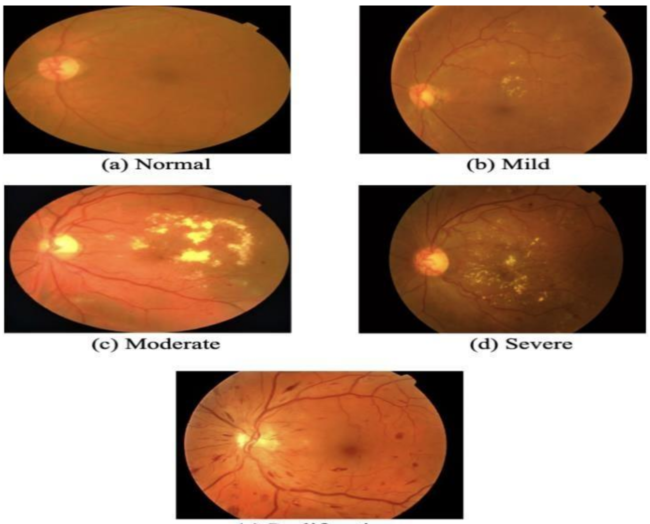
Keywords: DR, Fundus Image, Deep Learning, Neural Networks, DenseNet121.

Introduction

There are numerous eye conditions worldwide. One of the most chronic vision problems that most people confront is Diabetic Retinopathy (DR). DR is a micro vascular consequence of diabetes that affects the retina and is the cause of vision loss in people of working age. Retina is a light sensitive tissue present at the back of the eye that helps in transmitting various signals. High blood sugar in diabetic people can result in damage to the vascular system of the retina, resulting in the development of DR. There are two different categories of DR viz., proliferative and Non proliferative, as shown in Figure 1. Non proliferative Diabetic Retinopathy (NPDR) is the early stage of DR, making the tiny vessels of blood leak and the retina swell. They are further labeled into 3 types: Mild NPDR, Moderate NPDR, Severe NPDR, etc., on the basis of micro aneurysms, hard exudates, and haemorrhages. The most advanced stage of diabetic retinopathy is proliferative Diabetic Retinopathy (PDR), can be identified by the development of new blood vessels that emerge on the retina and in the vitreous humor. These brand-new blood vessels are extremely delicate and prone to bleeding, which can cause permanent vision loss and blindness. Diabetes identification and control, as well as regular eye exams, are critical for preventing the start and progression of diabetic retinopathy.[1]

In recent years, researchers have looked into convolutional neural networks (CNNs) as a viable method for automatically identifying DR in fundus image. CNNs are a sort of deep learning algorithm that can learn to recognise patterns in images. They have proven to be extremely effective in a variety of image analysis tasks, including object recognition, image segmentation, and image classification. Because of CNNs' ability to learn to distinguish patterns in images, models that can effectively identify DR have been developed. CNNs can also be used for remote screening in telemedicine systems, making them a promising DR detection method. However, there is still room for advancement in terms of CNN's sensitivity and specificity. The goal of this study is to investigate the use of CNNs for identifying DR in fundus images and to improve the model's performance by employing various strategies. This research will contribute to the field by developing strategies to increase the performance of CNNs for DR diagnosis, which will eventually lead to more efficient and effective screening for this deadly disease.[3]

Figure 1. The 5 grades of DR.



(a) Normal (b) Mild (c) Moderate (d) Severe (e) Proliferative

Results and Discussion

In this section we cover Proposed work, Datasets, Densnet121 Architecture and Layers, methods for risk prediction.

PROPOSED WORK

Our proposed approach aims to combat the various difficulties and disadvantages present within the existing approaches, where our aim is to perform image classification as accurately as possible; hence, we have used the Densenet121 CNN model. It is a 121-layer deep neural network that was trained on the ImageNet dataset for picture classification. DenseNet-121 has demonstrated cutting-edge performance on a variety of computer vision applications, including image classification, object detection, and semantic segmentation. DenseNet-121's architecture is distinguished by dense connectivity between layers. Each layer in a typical convolutional neural network receives as input the output of the preceding layer. Each layer in DenseNet-121 takes as input the feature maps from all preceding layers in the network. Because of the dense connection, the architecture is compact and efficient, and it can be trained with less parameters than typical CNNs.

Also the condition of the fundus image is analyzed based on the microaneurysms, cotton wool spots, hard exudates, hemorrhage, soft exudates, etc. and gives accurate results on the proposed model based on the techniques we use. Below, you can observe different CNN architectures performance over DR detection.

DATASET

In the proposed work, Densnert121 CNN model is executed to uprise the Accuracy. The only datasets with at least a thousand photos each are MESSIDOR-1, MESSIDOR-2, DDR, and EYEPACS. The dataset used for this paper is been extracted from kaggle. The EYEPACS dataset is considered which contains of 35,128 color fundus images. The selection of this dataset was based on its large size and the inclusion of high-resolution photos. Images are also collected from various kinds of cameras. The photos in the dataset are divided into five distinct classes. The 5 classes represent the severity level of DR, starting from No DR, Mild DR, Moderate DR, Severe DR and PDR.[7]

DENSENET121 ARCHITECTURE

- In the first layer, we apply convolutions to a kernel of size 7 X 7 and with a stride value that equals to 2. This results in an output of size 112 x 112 from 224 x 224.
- The first layer is followed by a pooling layer which involves a kernel of size 3 x 3 and with a stride value of 2, which results in an output of size 56 x 56.
- Then, Dense Block(1) contains 12 layers in which, 2 layers (Kernel[1] 1 x 1, kernel[2] 3 x 3) are repeated for 6 times.
- The follow up architecture consists of a transition layer(1) that comprises of a 1 x 1 convolutional layer and a 2 x 2 average pool layer with a stride value of 2, resulting in an output of size 28 x 28.
- Dense Block(2) contains double the layers of Dense Block(1).
- The second transition layer is applied here, in order to reduce the output size from 28 x 28 to 14 x 14.
- Two separate layers are applied before the final layer, which include two dense blocks and a third transitional layer.
- Finally, a classification layer is applied that classifies the input image of size 7 x 7.[6]

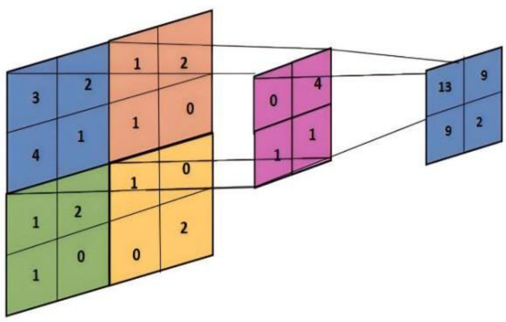
Table 1. DenseNet121 Architecture

Layers	Output size	DenseNet-121
Convolutional Pooling	112 x 112 56 x 56	7 x 7 conv, stride 2 3 x 3 max pool, stride 2
Dense Block (1)	56 x 56	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer (1)	56 x 56 28 x 28	1 x 1 conv 2 x 2 average pool, stride 2
Dense Block (2)	28 x 28	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$
Transition layer (2)	28 x 28 14 x 14	1 x 1 conv 2 x 2 average pool, stride 2
Dense Block (3)	14 x 14	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$
Transition Layer (3)	14 x 14 7 x 7	1 x 1 conv 2 x 2 average pool, stride 2
Dense Block (4)	7 x 7	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$
Classification Layer	1 x 1	7 x 7 global average pool fully-connected, softmax

CONVOLUTIONAL LAYER:

The convolution layer is a critical component of CNN. It is in charge of the majority of the network's processing load. This layer performs a scalar product between two matrices, as shown in below Figure 2, one of which is the set of learnable parameters known as a kernel, and the other is the restricted area of the receptive field. The kernel has more depth than an image but is smaller in size. As a result, if an image contains three RGB channels, the kernel height and width will be limited, but the depth will contain all three channels or any combination of the three.[2]

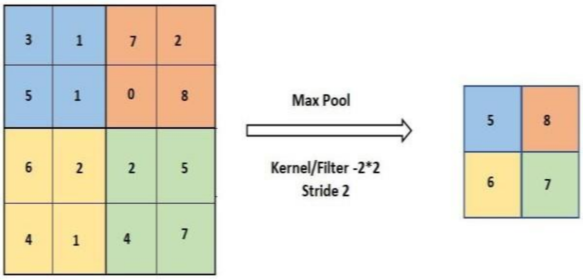
Figure 2. Convolution layer



POOLING LAYER

Pooling layers are interspersed between the convolutional layers in the dense blocks of DenseNet121. These pooling layers help reduce the spatial dimensions (width and height) of the feature maps while retaining important features, as depicted in Figure 3. Typically, max pooling or average pooling operations are used, where the maximum or average value within a pooling window is selected. This down sampling operation helps in capturing the most salient features while reducing the computational complexity of the network.

Figure 3. Pooling layer



FULLY CONNECTED LAYER:

At the end of the convolutional and pooling layers, a fully connected layer is typically added. This layer serves as the classification layer, responsible for mapping the extracted features to the appropriate output classes. The feature maps from the preceding layers are flattened into a 1-dimensional vector and fed into the fully connected layer.[4] The fully connected layer consists of neurons, each connected to all neurons in the previous layer. These connections enable the network to learn complex relationships between the features and the target classes. The output of the fully connected layer is passed through an activation function, such as softmax, to produce the final class probabilities.

IMPLEMENTATION

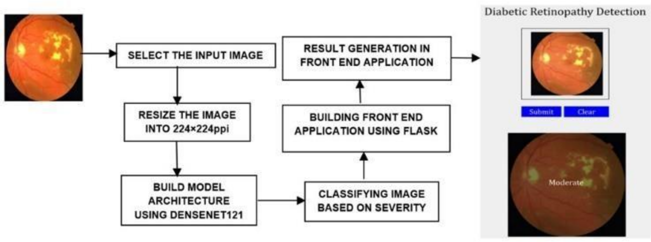
Our research paper is implemented using a three-module system, with each module being executed sequentially in different phases, as illustrated in Figure 4. The first module involves creating a Convolutional Neural Network (CNN) model to classify diabetic retinopathy. This module comprises the following steps:

1. Initially, we begin by resizing the image resolution to the desired input format, specifically 224 x 224 pixels per inch.
2. One effective strategy for model construction is transfer learning, which entails adapting pre-existing models to new tasks or datasets.
3. Our main focus revolves around utilizing the DenseNet121 architecture to build a machine learning model, incorporating essential components like the DenseNet average pooling layer and dropout layer.
4. To develop the DenseNet121 model, we load the pre-trained weights and remove the top classification layer. Subsequently, we add a global average pooling layer, followed by a fully connected layer with a dropout rate of 0.5.
5. We introduced a classification layer that classifies a given input image into 5 different classes. This architecture can be fine-tuned by updating the weights of the newly added layers while keeping the pre-trained weights unchanged.
6. The model is saved in ".h5" format, utilizing the "sigmoid" activation function within the dense layer.
7. By utilizing the model, we can categorize images into five severity-based categories.

During the second phase, the conversion of base64 to PIL (Python Image Library) takes place. This phase involves the following steps:

1. Working with a Python file that includes libraries such as Base64, Pillow, Numpy, and IO.
2. Converting the image data from base64 format to a PIL image by utilizing a base64 decoding function, such as **base64.b64decode()**. This decoding process converts the base64-encoded image data back to its original binary form.
3. Creating a BytesIO object using the **io.BytesIO()** function. This object acts as a buffer to hold the binary image data.
4. Loading the image is accomplished by employing the **Image.open()** function from the PIL library. The BytesIO object is passed as a parameter to load the image from the binary data.
5. Once the image is successfully loaded as a PIL Image object, various image processing operations can be performed according to specific requirements. These operations may include resizing, cropping, or applying filters to the image.[5]

Figure 4. Workflow of Diabetic Retinopathy Classification



Finally, in the third phase, the following steps are performed:

1. Working on a Python file that includes Flask, Tensor flow, PIL, and pre-processing utilities.
2. Creating a Flask web application and loading the trained model.
3. Using the CNN model to categorize the severity of diabetic retinopathy and estimate its severity grade.
4. Executing the Flask application, which displays a web application.

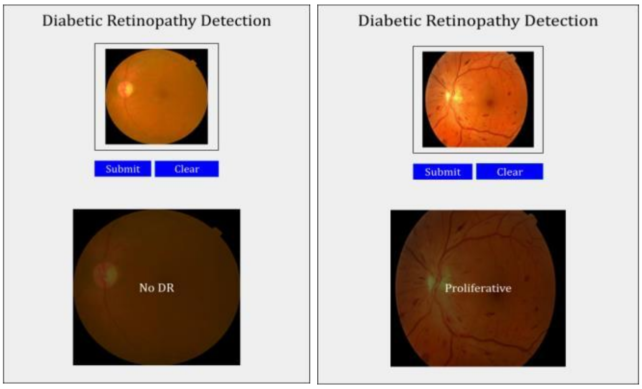
EXPERIMENTAL RESULTS

By the end of this paper, we have developed a web-based application that utilizes a digital fundus image of the eye to determine a patient's DR status, as demonstrated in Figure 5. This tool represents an improvement over traditional diagnostic methods for identifying DR as it only requires a digital fundus image. The implemented approach has yielded promising results, showcasing the effectiveness of CNNs in accurately categorizing the severity of DR.

The main objective of this paper was to design a rapid analysis system that utilizes deep learning techniques to automate the classification process of diabetic retinopathy. By training a deep CNN model on a large dataset of digital fundus images, the researchers achieved outstanding performance in diagnosing and grading the severity of DR.

The results obtained from the implemented method demonstrated high accuracy in categorizing retinopathy severity levels, providing crucial insights for medical professionals to make informed decisions regarding patient treatment and management. The rapid analysis capabilities of the deep CNN model open up possibilities for efficient screening programs, reducing the workload on ophthalmologists and improving the overall effectiveness of DR diagnosis.

Figure 5. Healthy(No DR) and a non healthy(proliderative DR) retinal image.



Furthermore, comparative assessments were conducted in the study to highlight the superiority of the proposed deep CNN-based method over traditional techniques for retinopathy classification. The findings clearly demonstrated that the deep CNN model surpassed conventional methods, underscoring the potential of deep learning in advancing medical image analysis and diagnosis.

Conclusion

In conclusion, our paper presented the successful implementation of a three-module system for classifying diabetic retinopathy. By employing transfer learning and the DenseNet121 architecture, we developed a robust machine learning model that proved to be highly effective. The integration of base64 to PIL conversion and the development of a Flask web application enabled efficient image processing and provided a user-friendly platform for accessing the model's predictions.

Our research showcased the potential of advanced machine learning techniques and web technologies in the field of medical image analysis and diagnosis. The accurate categorization of diabetic retinopathy severity levels holds significant importance for early detection and effective treatment planning. The outcomes of our study open doors for further research and practical applications in the medical field, where automated image analysis can support healthcare professionals in making informed decisions and enhancing patient care.

References

[1] Alyoubi, W. L., Shalash, W. M., & Abulkhair, M. F. (2020, January 1). Diabetic retinopathy detection through deep learning techniques: A review. Informatics in Medicine Unlocked. Elsevier Ltd. https://doi.org/10.1016/j.imu.2020.100377

[2] Tama, B.A., Vania, M., Lee, S. et al. Recent advances in the application of deep learning for fault diagnosis of rotating machinery using vibration signals. Artif Intell Rev 56, 4667–4709 (2023).https://doi.org/10.1007/s10462-022-10293-3

[3] Oh, K., Kang, H.M., Leem, D. et al. Early detection of diabetic retinopathy based on deep learning and ultra-wide-field fundus images. Sci Rep 11, 1897 (2021).https://doi.org/10.1038/s41598-021-81539-3

[4] Roychowdhury S, Koozekanani DD, Parhi KK. DREAM: diabetic retinopathy analysis using machine learning. IEEE J Biomed Health Inform. 2014 Sep;18(5):1717-28. doi: 10.1109/JBHI.2013.2294635. PMID: 25192577.

[5] A.Hashim, Fatma & Salem, Nancy & Seddik, Ahmed. (2013). Preprocessing of Color Retinal Fundus Images. 10.1109/JEC-ECC.2013.6766410.

[6] Huang, Gao & Liu, Zhuang & van der Maaten, Laurens & Weinberger, Kilian. (2017). Densely Connected Convolutional Networks. 10.1109/CVPR.2017.243.

[7] https://www.kaggle.com/competitions/diabetic-retinopathy-detection/data