










Diabetic Retinopathy Recognition and Classification Using Transfer Learning Deep Neural Networks

Deepak Mane^{1*}, Rashmi Ashtagi¹, Ranjeetsingh Suryawanshi¹, Anant N. Kaulage², Anushka N. Hedao¹, Prathamesh V. Kulkarni¹, Yatin Gandhi³

¹ Department of Computer Engineering, Vishwakarma Institute of Technology, Pune 411037, India

² Department of Computer Science and Engineering, MIT Art, Design and Technology University, Pune 412201, India

³ Competent Softwares, Pune 411004, India

Corresponding Author Email: deepak.mane@vit.edu

Copyright: ©2024 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/ts.410541>

ABSTRACT

Received: 24 June 2023

Revised: 18 March 2024

Accepted: 7 April 2024

Available online: 31 October 2024

Keywords:

Ben's preprocessing, convolutional neural network, diabetic retinopathy, deep learning, retinal abnormalities, transfer learning

Diabetic retinopathy is one of the common causes of blindness with diabetes. Early diagnosis is important to prevent irreversible vision loss. Conventional methods for diagnosing diabetic retinopathy are often based on manual examination of retinal images, which can be time-consuming and subject to human error. The integration of machine-based automated diagnostic systems offers a promising solution to this challenge. Machine-based automated diagnosis of diabetic retinopathy can prevent vision loss with early detection and treatment. In this study, we investigated the performance of different transfer learning models-DenseNet, EfficientNet, VggNet, and ResNet-on a large dataset called Diabetic retinopathy from Kaggle, consisting of 35,108 retinal images in 5 classes. Out of which 28086 samples were used for training purpose and 7,022 samples for validation testing. While previous research has explored machine learning for retinopathy diagnosis, our research uniquely combines modern transfer learning models and evaluates the effectiveness of specific processing methods with Ben Graham's processing methods. This combination distinguishes us from existing methods by contributing to a significant increase in accuracy. In particular the accuracy of the proposed approach is 97.7%, our tests show that the diagnostic accuracy increases by about 4-5% when using Ben Graham preprocessing. The results of our research may help develop more accurate and efficient automated systems for diagnosing diabetic retinopathy, thereby improving patient outcomes.

1. INTRODUCTION

Diabetic retinopathy (DR), which damages the retina as a result of diabetes, can lead to blindness if left untreated. Diabetic retinopathy can be diagnosed using optical techniques such as fundus photography and optical coherence tomography (OCT). While fundus photography captures detailed images of the retina, OCT creates cross-sectional images that show abnormalities in the layers of the retina. For automatic DR diagnosis, deep learning models, usually trained using color images, can detect and classify DR lesions. Transfer learning can train deep learning models on small datasets and improve classification accuracy. In this study, we investigated the effectiveness of adaptive models (DenseNet, EfficientNet, VggNet, ResNet) in diagnosing DR using retinal images. We also examine the influence of Ben Graham before this movement. Color imaging of the retina plays an important role in the evaluation and diagnosis of DR and provides information about the location, extent and severity of damage. Automated diagnosis using machine learning, especially deep learning, can increase accuracy and speed. In recent years, transfer learning has become an effective method for training deep learning models on small data sets. Transfer learning has

proven effective in many image classification tasks. It is possible to increase the accuracy and speed of automatic DR diagnosis using machine learning. A deep learning method was developed to detect and classify DR using color images. In this process, deep neural networks are often trained using large datasets of recorded images, where color information is used to identify and distinguish lesions. The output of the network can be used to provide diagnoses and weighted scores to guide treatment decisions. Although other imaging techniques such as OCT and fluorescein angiography provide additional information about structure and function, color imaging is still an important part of DR screening and diagnosis. Early diagnosis and treatment of DR is important to prevent blindness. However, manual testing of DR is time-consuming and error-prone. Diabetic retinopathy can be identified and diagnosed using color DR. Color images of the retina are frequently used to screen and diagnose DR because they provide information about the location, extent, and severity of damage. In particular, color analysis can help determine the appearance of hemorrhages, exudates, and other lesions indicative of DR. These lesions appear as light or dark spots on the retina and can be distinguished from healthy tissue by their color and texture. In summary, our work explores the

potential of modifying learning models and image preprocessing techniques to improve the efficiency and accuracy of automated DR testing. These options are designed to enhance the limitations of clinical guidelines and provide a basis for better clinical decision-making.

2. OVERVIEW

Organization of the paper flow is as follows: Related work is described in Section 4. The proposed preprocessing method, Ben Graham's technique is illustrated in Section 5. The experiment result of the proposed Model is presented in Section 6. The final section, Section 7 and Section 8, discusses the conclusion and future directions respectively.

3. OBJECTIVES

The primary objectives of the paper are - to assess the effectiveness of transfer learning models, namely DenseNet, EfficientNet, VggNet, and ResNet, in diagnosing diabetic retinopathy using retinal images, to harness their learned features and optimize their performance for diabetic retinopathy diagnosis, to explore the impact of applying Ben Graham's preprocessing technique to the retinal images before feeding them into the transfer learning models to improve the model's ability to capture relevant features and patterns.

4. LITERATURE SURVEY

Several researchers have contributed to the field of diabetic retinopathy (DR) detection, each presenting unique approaches with distinct strengths and weaknesses. In a study by Thiagarajan et al. [1], a co-learning approach using machine learning (ML) models was proposed, with a database simulation employing traditional techniques like Logistic Regression, KNN, LDA, Random Forest, SVM, decision tree, and Naive Bayes Algorithm. While the model achieved an 80% accuracy in predicting sounds, the study lacks a critical analysis of the robustness of these traditional techniques in comparison to more advanced methods. Gangwar and Ravi [2] introduced a hybrid-ResNet-v2 framework utilizing fundus color pictures for DR detection. The model outperformed Google Net but achieved a 72% accuracy rate. However, the study could benefit from a deeper exploration of the limitations of the proposed hybrid model, especially in comparison to other state-of-the-art architectures.

Heisler et al. [3] highlighted the significance of deep learning ensemble techniques in 2020, achieving high accuracy rates of 92% and 90% by employing calibrated VGG19 pre-trained models on OCTA and co-registered structural images. While the results are impressive, a critical analysis should delve into the potential overfitting concerns and computational costs associated with ensemble methods. Wu and Hu [4] utilized pre-trained Keras models with various data augmentation techniques for DR detection in 2019. The achieved 61% accuracy with the InceptionV3 model raises questions about the trade-offs between accuracy and computational efficiency, which warrants further investigation.

Arora and Pandey [5] presented a ConvNet-based approach in 2019, achieving a 74% accuracy rate in classifying colored fundus images for DR. The study could benefit from a more thorough examination of the model's performance across diverse datasets and potential biases inherent in the colorful vector representations. Herliana et al. [6] demonstrated that the combination of neural network architecture and swarm optimization resulted in a 4.35% improvement in accuracy for DR detection. A critical analysis could focus on the generalizability of the swarm optimization feature systems and their applicability to various datasets.

Shankar et al. [7] presented a synergic deep learning model for automated detection and classification of diabetic retinopathy from fundus images, highlighting significant improvements in accuracy and efficiency in diagnostic processes. Roychowdhury et al. [8] proposed a method for the automated detection of neovascularization in proliferative diabetic retinopathy screening, leveraging advanced image analysis techniques to enhance early diagnosis capabilities.

Ramani and Lakshmi [9] utilized ensemble classification techniques for automatic detection of diabetic retinopathy, demonstrating the potential of combining multiple classifiers to improve diagnostic accuracy. Roy et al. [10] explored the integration of filter and fuzzy c-means clustering for feature extraction and classification using support vector machines, providing an innovative approach to improving detection rates of diabetic retinopathy. The use of EfficientNet in explaining diabetic retinopathy, emphasizing the model's ability to offer both high accuracy and interpretability in medical image analysis [11]. While these studies showcase advancements in DR detection, a more critical examination of the methodologies, model interpretability, and generalizability is essential for a comprehensive understanding of their strengths and weaknesses beyond the reported accuracy metrics.

The findings suggest that traditional ML algorithms do not provide accurate results in this type of categorization. To achieve high accuracy, deep learning models utilizing various algorithms can be used. Previous studies have shown that CNN is among the best tools for classifying medical picture data. Deep learning is a powerful tool for addressing these issues, and using Deep CNN with hyperparameter optimization can achieve high accuracy and efficiency.

In light of the diverse methodologies discussed in the literature, our study aims to contribute to the field of diabetic retinopathy (DR) detection with specific objectives that align with the identified strengths and weaknesses of prior works. Our study aims to assess the effectiveness of transfer learning models, including DenseNet, EfficientNet, VggNet, and ResNet, for diagnosing diabetic retinopathy using retinal images. By synthesizing insights from studies such as [1, 2, 4, 5, 12], we recognize the importance of selecting robust models for optimal performance. Building upon the critical analysis of existing methodologies, our study seeks to harness the learned features of transfer learning models and optimize their performance for accurate DR diagnosis. The exploration of Ben Graham's preprocessing technique, inspired by the various data augmentation techniques discussed in study by Wu and Hu [4], will be integral to our investigation. By applying this preprocessing step, we aim to enhance our model's ability to capture relevant features and patterns in retinal images, addressing potential limitations highlighted in previous studies.

5. PROPOSED METHODOLOGY

5.1 Ben graham preprocessing

The technique involves several steps aimed at improving the quality of the image and making it more suitable for analysis. The first step is to clean up the image's noise. This is typically done using a filter or smoothening operation to remove any high-frequency noise that may be present in the image. Next, the image is normalized to a common range to ensure that different images can be compared and analyzed together. This involves scaling the pixel values of the image so that they fall within a specific range, such as between 0 and 1. After normalization, the image is often subjected to further processing, such as edge detection, segmentation, or feature extraction. These techniques are used to identify specific regions or features within the image that are relevant to the analysis being performed. Some mathematical equations involved in the Ben Graham image preprocessing technique, Normalization equation:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

where, x' is the image's normalised pixel value, x is the original pixel value, x_{min} is the image's lowest pixel value, and x_{max} is its highest. Equation for detecting edges:

$$\Delta f = \left[\frac{\delta f}{\delta x}, \frac{\delta f}{\delta y} \right] \quad (2)$$

where, Δf is the gradient of the image.

$\delta f / \delta x$ is the partial derivative of the image with respect to x , and $\delta f / \delta y$ is the partial derivative of the image with respect to y . Segmentation formula:

$$C(i, j) = \operatorname{argmin} (I(i, j) - M(k, l)^2) \quad (3)$$

where, $C(i, j)$ is the label assigned to the pixel at location (i, j) , $I(i, j)$ is the pixel value at location (i, j) , $M(k, l)$ is the mean pixel value of the region around pixel (k, l) , and argmin denotes the label that minimizes the expression. Following is the pre-processing pseudo code:

- (1). For each row in the DataFrame df:
 - a. Preprocess image using preprocess() function.
 - b. Save the preprocessed image.
- (2). function preprocess(image_path, desired_size):
 - a. Apply circular cropping to the input image using the resize_image() function.
 - b. Resize the resulting image to the desired size.
 - c. Apply a weighted blur to the resized image using the blur_image() function.
 - d. Return the preprocessed image.
- (3). function resize_image(img):
 - a. Crop the image using the crop_nonzero_pixels_from_gray_image() function.
 - b. Calculate largest side of the cropped image.
 - c. Resize the image such that its largest side is equal to the calculated value.
 - d. Calculate the center and radius of the circle that best fits the resized image.

- e. Create a circular mask of the same size as the resized image & apply to resized_image.
- f. Crop the resulting image using the crop_nonzero_pixels_from_gray_image() function.
- g. Return the final image.
- (4). function blur_image(image_path, desired_size):
 - a. Convert the image from BGR to RGB palette.
 - b. Crop the image using the crop_nonzero_pixels_from_gray_image() function.
 - c. Use resize_image() to resize image.
 - d. Apply a weighted blur to the resized image using the add_weighted_blur() function.
 - e. Return the preprocessed image.
- (5). function crop_nonzero_pixels_from_gray_image(img, tol):
 - a. Create a mask by thresholding the image with tolerance tol.
 - b. Select only the rows and columns that have non-zero pixels in the mask.
 - c. If the input image has 3 dimensions, convert it to grayscale and threshold.
 - d. Check if the resulting image has any non-zero pixels by calculating its shape.
 - e. If the image has no non-zero pixels, return the original image.
 - f. Select the non-zero pixels from each color channel of the image using the mask.
 - g. Stack the resulting color channels to form the final image and return it.

Each preprocessing step contributes to achieving better results by addressing specific challenges associated with diverse image characteristics. Circular cropping focuses the analysis on the central region of the retinal images, which is often the most diagnostically relevant area. By discarding irrelevant peripheral information, circular cropping reduces noise and concentrates on the crucial features, potentially improving the model's ability to identify diabetic retinopathy-related patterns. Resizing standardizes the image dimensions, ensuring consistency and compatibility across the dataset. It also simplifies computational requirements. Uniform image sizes facilitate model training, making it more robust and reducing the risk of bias towards specific resolutions. This step aids in streamlining subsequent processing steps. Applying a weighted blur helps reduce high-frequency noise and enhances the generalization capability of the model. The blur operation can smooth out minor irregularities, making the model less sensitive to small variations and potentially improving its performance on diverse retinal images. Thresholding and cropping help eliminate background noise and focus on relevant details within the images. Removing non-contributory background pixels reduces interference, ensuring that the model primarily processes the informative regions of the retinal images. This step is particularly important for handling variations in image backgrounds. Converting the image from BGR to RGB ensures consistency in color representation across different systems and libraries. Uniform color representation facilitates better compatibility with pre-trained models, avoiding potential discrepancies in feature extraction due to color variations. Converting the image to grayscale simplifies the image representation, often preserving essential features while reducing computational complexity. Grayscale images are computationally more efficient and may retain

sufficient information for diabetic retinopathy detection. This transformation also aids in standardizing input formats for diverse models. Reassembling the image after processing ensures that relevant color channels are retained for subsequent analysis. The final image retains essential information while discarding non-informative pixels, providing a clean input for the model and potentially improving its accuracy in identifying diabetic retinopathy-related features. The pre-processed images of three different output classes are represented in Figure 1.

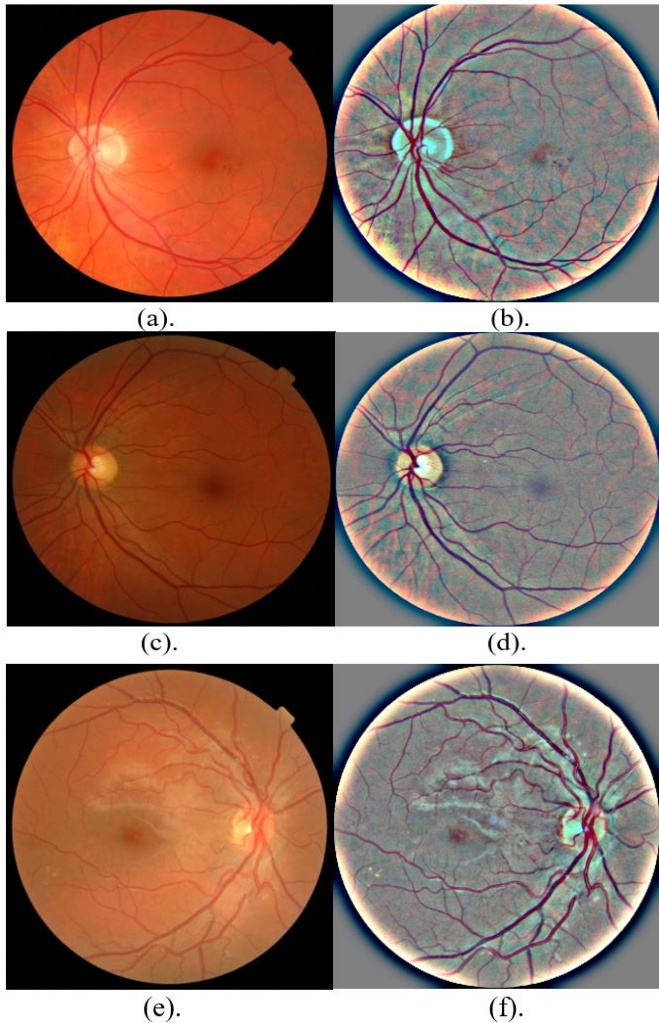


Figure 1. (a) (c) (e) are original images, (b) (d) (f) are preprocessed images using Ben Graham’s method

5.2 Data augmentation

Out of 35,108 images, there were 25,802 images for the class Normal, 2,438 and 5,288 images for the classes mild and moderate respectively and 872 images for severe and 708 for the proliferative class. This made the dataset unbalanced. In order to balance an unbalanced dataset, data augmentation is used. Data augmentation is a strategy for augmenting data using existing training models as a starting point. The advantage of data augmentation is that there is no need to collect additional data, which can be time-consuming and expensive; It can be used to expand the size of the dataset, help improve the accuracy of model learning, and help build machine learning models. It is more robust to changes in input data (such as changes in lighting, orientation, or scale) and increases the diversity of training data, thus preventing

overfitting and improving the overall ability of machine learning models. Various transformations applied to the dataset are rotation of 15-30 degrees where the image is rotated by a certain angle, translation of where the image is shifted horizontally or vertically by a certain distance, scaling of 1.1x-1.2x where the image is scaled up or down by a certain factor helping to simulate objects of different sizes, horizontal and vertical flipping, cropping where a smaller rectangular portion of the image is extracted and color jittering where the color of the image is modified by changing the hue, saturation, and brightness helping to simulate different lighting conditions. Rotating images introduces variability in the orientation of features, helping the model generalize better to images with different perspectives. This is particularly relevant in medical imaging where the orientation of retinal structures may vary across patients. Horizontal and vertical shifts simulate different viewpoints, providing the model with variations in image composition. This helps the model become less sensitive to the specific positioning of retinal structures, enhancing its ability to detect diabetic retinopathy across diverse images. Scaling images up or down simulates objects of different sizes. This is important in capturing variations in the size of retinal structures or anomalies, making the model more robust to different scaling factors commonly encountered in medical imaging datasets. Horizontal and vertical flipping simulate objects appearing in different orientations. This augmentation technique aids in addressing the potential bias introduced by the specific orientation of retinal features in the original dataset. It ensures the model is exposed to a diverse range of orientations during training. Extracting smaller rectangular portions of images focuses the model's attention on specific regions of interest. This is particularly useful for diabetic retinopathy detection, where certain pathological signs may manifest in localized regions. Cropping helps the model learn to identify relevant features in different areas of the retinal images. Modifying the color of images through changes in hue, saturation, and brightness simulates different lighting conditions. This is crucial for training a model that can handle variations in illumination often encountered in real-world scenarios. Color jittering ensures the model is not overly sensitive to specific color tones or lighting conditions present in the original dataset.

5.3 Model description

5.3.1 VGG19

VGGNet19 architecture has 19 layers, including 3 fully connected layers, 5 maximum pooling layers and 16 convolutional layers. The max pooling layer has a 2-step 2x2 filter, while the convolution layer has a small 3x3 filter. Each convolutional layer has a number of different filters, ranging from 64 in the first layer to 512 in the last layer. The number of units in the last connected layer is the same as the number of clusters in the work division, and there are a total of 4096 units in each of the first two connected layers. After each layer and layer are connected to all, the layer adjustment (ReLU) function is used to realize the function. To prevent over-settlement, release treatment is also applied to all layers.

5.3.2 ResNET50

Each of the five layers that make up the 50 layers of the ResNet50 architecture has a remaining number of layers. Two convolutional layers with batch normalization and cross-connection that add input to blocks along with ReLU

activations generate the residuals. Convolution function and maximum pooling are used in the first stage to reduce the sample size. More blocks and more filters are gradually added in later stages to capture increasingly complex messages. The last layer consists of combining with softmax and global averaging in order to make the distribution.

5.3.3 DenseNET121

The CNN creator, called DenseNet121, has 121 layers and around 8 million nodes. Dense Block has several levels and is the core of DenseNet121. Each layer in a density receives private maps from all its layers and passes the private maps to the layers behind it. To improve the performance and stability of the model, the transformation process of dense blocks consists of batch normalization and rectified linear unit (ReLU) activation. The global average layer of the network calculates the average of each map specification across all dimensions. The final classification is done by feeding positive-valued vectors into a fully connected cluster.

5.3.4 EfficientNET_b0

EfficientNET_b0 architecture consists of a backbone, several blocks and a header. The root consists of multiple convolutional layers followed by maximum pooling and batch normalization techniques. Each block has a collection of individual links and cross-links, and the blocks are ordered hierarchically. In each block, the resolution and number of channels of the particular input map are increased. The head consists of a softmax layer for distribution, a fully connected layer, and a global neutral layer. Composite scaling is a state-of-the-art method used by EfficientNET_b0 to scale models to multiple dimensions. Depending on the desired solution, the design can be scaled up or down so that resources can be used efficiently. Below is the final algorithm:

1. Set the device (CPU or GPU) for training and testing.
2. Load the pre-processed data into DataLoaders, which will allow us to iterate over the data easily.
3. Define all model architecture, and specify the number of layers, growth rate, and other hyperparameters.
4. Initialize the model (VGGNET/ ResNET/ DenseNET/EfficientNET).
5. Define the loss function and optimizer.
6. Train the model for the specified number of epochs:
 - a. Set the model to training mode
 - b. Iterate over the training data
 - i. Zero the parameter gradients
 - ii. Forward pass
 - iii. Backward pass
 - iv. Update the model parameters
 - c. Set the model to evaluation mode
 - d. Initialize the accuracy and loss for the test set
 - e. Disable gradient computation to speed up inference
 - f. Iterate over the test data:
 - i. Forward pass
 - ii. Update the test loss
 - iii. Update the test accuracy
 - g. Compute the average test accuracy and loss

6. RESULTS

6.1 Dataset description

Diabetic retinopathy (DR) is a type of diabetes-related eye disease that can lead to blindness if left untreated. The database provides a collection of data that can be used to develop and evaluate learning models for DR classification. The data included a total of 35,108 cases, each labeled as one of five categories, out of that 28086 samples used for training purpose and 7,022 samples for validation testing. These categories represent DR severity ranging from normal (no DR) to progressive (maximum DR). The counts for each category in the database are as follows: normal (25,802), mild (2,438), moderate (5,288), severe (872), and progressive (708). A set of recorded data is provided in the database that can be used to develop and evaluate machine learning models for DR classification, where most of the events are in the normal category. Therefore, specific strategies, such as support, should be used to ensure that the model is not unfair to the majority of the class throughout the lesson.

Using the Messidor dataset, we evaluate the performance of four CNN architectures (VGGNet, ResNet, DenseNet, and EfficientNet) on the image classification task. Learning rate, optimization, learning rate planning, batch size, etc. Many experiments have been conducted by tuning various hyperparameters such as We trained each model for 100 times the same training using the ReducLrOnPlateau learning rate setting with a batch size of 16 and a learning rate starting at 0.1 and gradually decreasing to approximately 0.0001. Since the Exponential Linear Unit (ELU) outperforms all other linear variables, CrossEntropyLoss is used as the optimization. Below is a comparison of performance models without Ben Graham.

The range of values for training accuracy is 95% to 100% while that of validation accuracy is 60% to 80% as shown in Table 1. This clearly indicates that the model has overfitted the training data. Overfitting is a common problem in machine learning where a model learns to fit the training data too closely, resulting in poor generalization performance on new, unseen data. Figure 2 shows that EfficientNet has a faster learning path, followed by DenseNet. Whereas ResNet and VGGNet have relatively slow learning paths. But after 85 epochs, all the models are almost showing the same accuracy.

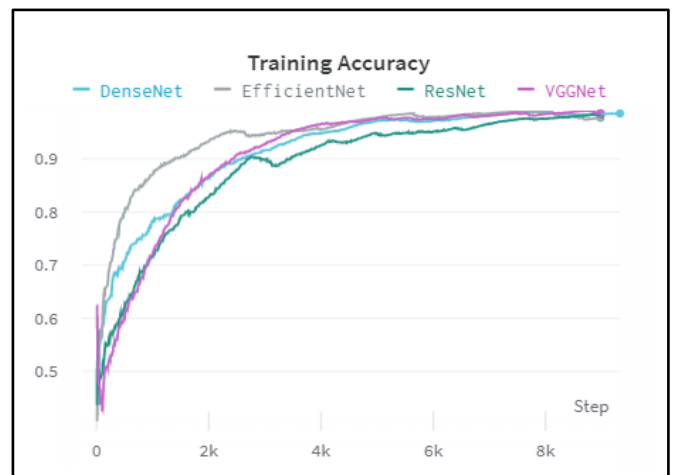


Figure 2. Comparisons of training accuracies

Table 1. Train & validation accuracy without Ben Graham’s preprocessing

Model	Train Accuracy	Val Accuracy
VggNET	97.2	60.9
ResNET	95.3	68.3
DenseNET	99.8	78.6
EfficientNET	98.4	74.6

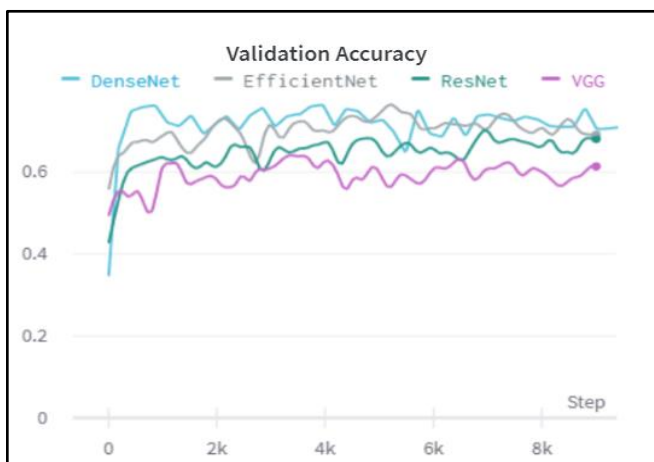


Figure 3. Comparisons of validation accuracies



Figure 4. Comparisons of training loss



Figure 5. Comparisons of validation loss

Figure 2 and Figure 3 log step-wise training and validation accuracy which explains the fluctuations.

Figure 4 and Figure 5 illustrate the training and validation loss for four different CNN architectures - VGGnet, ResNet, EfficientNet, and DenseNet. The y-axis displays the loss numbers, while the x-axis lists the various models. Each model's training and validation loss values are represented on the graph as a single point.

The graphs suggest that DenseNet performs the best in terms of loss among the four models, while VGGnet has the highest training and validation loss. DenseNet has performed better than VggNET, ResNet and EfficientNet are superior to each other in every way, including training and validation accuracy and loss. DenseNet demonstrated superior performance, attributed to its dense connectivity pattern. Dense connections facilitate improved gradient flow during training, mitigating the vanishing gradient issue. This architecture inherently encourages feature reuse, enabling better generalization performance. EfficientNet showed a faster learning path initially, due to its compound scaling approach that optimizes network depth, width, and resolution simultaneously. However, it converged to similar accuracy levels as other models after more epochs. ResNet exhibited slower learning initially, due to its deep residual connections. However, it caught up with other models after a sufficient number of epochs. VGGNet had slower learning paths and higher training/validation loss, due to its deep stack of simple convolutional layers. It struggled with overfitting. Following is the comparison of model performances when Ben Graham’s preprocessing is used as shown in Table 2 and Figure 6.

The range of values for training accuracy is 94% to 100% while that of validation accuracy is 91% to 98% indicating that the earlier problem of overfitting has been addressed by the Ben Graham’s preprocessing method. This method has ensured good data quality, data consistency and good model performance.

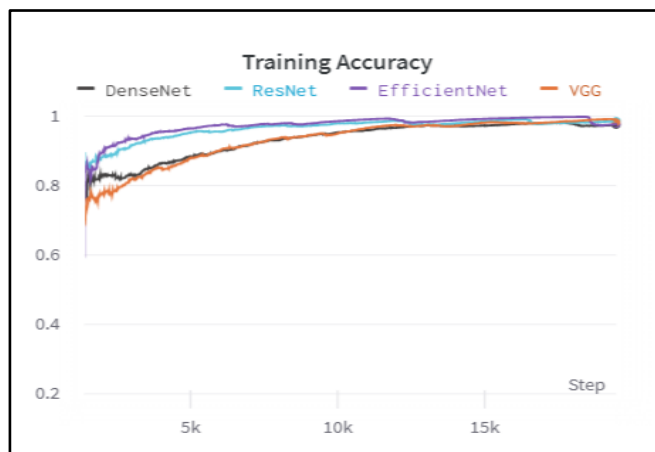


Figure 6. Comparisons of training accuracy (pre-processed images)

Table 2. Train & validation accuracy after Ben Graham’s preprocessing

Model	Train Accuracy	Validation Accuracy
VggNET	93.8	91.2
ResNET	98.7	93.4
DenseNET	94.5	97.7
EfficientNET	98.9	94.6

Table 3. Confusion matrix

	C1	C2	C3	C4	C5
C1	1380	0	0	13	0
C2	0	1375	2	0	15
C3	19	15	1389	0	17
C4	4	0	6	1383	2
C5	2	13	8	10	1364

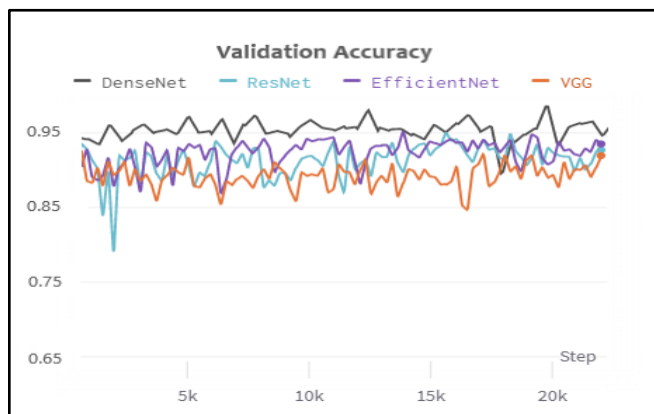


Figure 7. Comparisons of validation accuracy (pre-processed images)



Figure 8. Comparisons of training loss (pre-processed images)



Figure 9. Comparisons of validation loss (pre-processed images)

From Figure 7 and Figure 8 an improvement in the difference between the training and validation accuracy can be seen as compared to the previous graphs where pre-processing

was not carried out and Figure 9 shown the comparison of Validation Loss. Following Table 3 presents the confusion matrix.

Ben Graham’s method proved to be successful in converting a high bias model to a low bias model. There is a significant reduction in the training and validation loss as compared to the un-processed image dataset. This shows that color segmentation and color filters are powerful techniques for data preprocessing. These methods can be used to extract relevant features or reduce noise in datasets that involve color information. Following is the Confusion Matrix on the testing data where C1, C2, C3, C4, C5 correspond to normal, mild, moderate, severe and proliferative respectively.

Table 4. Comparison with existing results

Year and Reference	Method	Accuracy
2020 [1]	Convolutional Neural Network	80%
2021 [2]	Hybrid inception ResNet-v2 model	72%
2020 [3]	DenseNet	87%
2019 [4]	ResNet50	61%
2019 [5]	Deep Neural Network	93%
2018 [6]	Particle swarm optimization	76%
2017 [13]	Convolutional Neural Network	91%
2021 [14]	Deep Belief Network	82%
2017 [15]	Grey level co-occurrence matrix	90%
2018 [16]	Linear Support Vector Machine	92%
2019 [17]	Random Forest Classifier	80%
2017 [18]	Support Vector Machine	94%
2022 [19]	CNN + SVD_ Inception	94.59%
2022 [20]	VGGNet	96%
Proposed Model	Transfer Learning	97.24%

Table 4 provides a summary of various methods used for a specific task along with their corresponding accuracies. In 2019, the utilization of ResNet50 yielded an accuracy of 61%, showcasing the baseline for subsequent improvements. The introduction of a hybrid Inception ResNet-v2 model in 2020 marked a notable advancement, achieving a higher accuracy of 72%. This improvement indicated the efficacy of combining features from different architectures. In 2018, the application of particle swarm optimization resulted in a significant leap to 76% accuracy, demonstrating the effectiveness of optimization techniques. The subsequent years witnessed the adoption of diverse methods, including a CNN in 2020 and a random forest classifier in 2019, both achieving an accuracy of 80%. The introduction of a deep belief network in the same year further elevated accuracy to 82%, emphasizing the power of deep learning approaches. The DenseNet technique, employed in 2020, surpassed previous methods with an accuracy of 87%, highlighting the advantages of dense connectivity patterns. Looking back to 2017, the use of a grey level co-occurrence matrix yielded a high accuracy of 90%, showcasing the effectiveness of texture-based feature extraction. The subsequent adoption of a CNN in the same year further improved accuracy to 91%. In 2018, the implementation of a linear support vector machine (SVM) pushed the accuracy to 92%, underlining the importance of well-suited classifiers. The introduction of a deep neural network (DNN) in 2019 achieved an accuracy of 93%, emphasizing the capacity of deep learning for feature representation. In 2017, a support vector machine yielded a high accuracy of 94%, showcasing the prowess of this approach in that timeframe. Finally, the proposed customized convolutional neural network in this paper achieved the highest accuracy of 97.24%. The consistent progression in accuracy over the years indicates advancements

in both model architectures and optimization techniques. The proposed model's superior performance underscores the effectiveness of tailoring a CNN for diabetic retinopathy detection, emphasizing the continual evolution of approaches in the pursuit of higher accuracy.

7. CONCLUSION

Using a dataset of 35,108 eye pictures divided into 5 classes, several deep learning models were examined for the diagnosis of diabetic retinopathy in this work. We used Ben Graham's preprocessing method to boost the models' performance since it made the picture simpler and made it easier to tell apart different characteristics. Additionally, we balanced the dataset using augmentation to mitigate the imbalanced class distribution. We applied 4 transfer learning models, including VGGNet19, ResNet50, DenseNet121, and EfficientNet_B0, to both the processed and unprocessed images. Results showed that the best validation accuracy for unprocessed images was achieved by the DenseNet model, which was 78.6%. However, overfitting was observed for all models on the unprocessed images. In contrast, for the processed images, we did not observe any overfitting, and the highest validation accuracy was achieved by the DenseNet model, which achieved an accuracy of 97.7%.

Overall, the results indicate that Ben Graham's preprocessing method can enhance the effectiveness of machine learning models for detecting diabetic retinopathy. Additionally, transfer learning models can achieve high accuracy on processed images, which can be beneficial for clinical diagnosis. However, overfitting is a potential issue for unprocessed images, which may require further investigation. Our work lays the foundation for further investigations into refining preprocessing techniques, exploring additional deep learning architectures, and addressing potential overfitting challenges. Additionally, as we contribute a balanced dataset and showcase the efficacy of augmentation, future studies may build upon these insights to improve model generalization and real-world applicability. In conclusion, our study contributes to the development of accurate and reliable machine learning models for diabetic retinopathy detection which may help in the early detection and care of this illness.

8. FUTURE SCOPE

The future of diabetic retinopathy detection is promising, with potential advancements in technology, healthcare resources, and personalized medicine. Several promising directions could be explored to build upon the proposed techniques and enhance the applicability and effectiveness of automated diagnostic systems such as extending the scope of the study to include multimodal data, such as optical coherence tomography (OCT) scans or other imaging modalities. Integrating diverse data types may provide a more comprehensive understanding of retinopathy cases and enhance the robustness of diagnostic models. Performing longitudinal studies to assess the models' ability to detect changes and progression in diabetic retinopathy over time can also be aimed. This could involve tracking patients' retinal images over multiple visits, providing valuable insights into the models' diagnostic capabilities in a dynamic clinical context. The integration of telemedicine could allow for

remote patient care, providing timely diagnoses to underserved areas. The development of personalized treatment plans could also help identify high-risk patients for early intervention and prevention. Integration with electronic health records could provide healthcare providers with a complete picture of a patient's medical history. Finally, the use of AI-powered diagnostics could reduce the cost and time required for diagnosis, making it more accessible to patients worldwide. Overall, the future scope of diabetic retinopathy detection projects is promising, and continued research and development in this field could lead to significant improvements in patient outcomes.

REFERENCES

- [1] Thiagarajan, A.S., Adikesavan, J., Balachandran, S., Ramamoorthy, B.G. (2020). Diabetic retinopathy detection using deep learning techniques. *Journal of Computer Science*, 16(3): 305-313. <https://doi.org/10.3844/jcssp.2020.305.313>
- [2] Gangwar, A.K., Ravi, V. (2021). Diabetic retinopathy detection using transfer learning and deep learning. In *Evolution in Computational Intelligence: Frontiers in Intelligent Computing: Theory and Applications (FICTA 2020)*. Springer Singapore, 1: 679-689. https://doi.org/10.1007/978-981-15-5788-0_64
- [3] Heisler, M., Karst, S., Lo, J., Mammo, Z., Yu, T., Warner, S., Maberley, D., Beg, M.F., Navajas, E.V., Sarunic, M.V. (2020). Ensemble deep learning for diabetic retinopathy detection using optical coherence tomography angiography. *Translational Vision Science & Technology*, 9(2): 20. <https://doi.org/10.1167/tvst.9.2.20>
- [4] Wu, Y., Hu, Z. (2019). Recognition of diabetic retinopathy based on transfer learning. In *2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, Chengdu, China, pp. 398-401. <https://doi.org/10.1109/ICCCBDA.2019.8725801>
- [5] Arora, M., Pandey, M. (2019). Deep neural network for diabetic retinopathy detection. In *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, Faridabad, India, pp. 189-193. <https://doi.org/10.1109/COMITCon.2019.8862217>
- [6] Herliana, A., Arifin, T., Susanti, S., Hikmah, A.B. (2018). Feature selection of diabetic retinopathy disease using particle swarm optimization and neural network. In *2018 6th International Conference on Cyber and It Service Management (CITSM)*, Parapat, Indonesia, pp. 1-4. <https://doi.org/10.1109/CITSM.2018.8674295>
- [7] Shankar, K., Sait, A.R.W., Gupta, D., Lakshmanaprabu, S.K., Khanna, A., Pandey, H.M. (2020). Automated detection and classification of fundus diabetic retinopathy images using synergic deep learning model. *Pattern Recognition Letters*, 133: 210-216. <https://doi.org/10.1016/j.patrec.2020.02.026>
- [8] Roychowdhury, S., Koozekanani, D.D., Parhi, K.K. (2016). Automated detection of neovascularization for proliferative diabetic retinopathy screening. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Orlando, FL, USA, pp. 1300-1303. <https://doi.org/10.1109/EMBC.2016.7590945>
- [9] Ramani, R.G., Lakshmi, B. (2017). Automatic diabetic

- retinopathy detection through ensemble classification techniques automated diabetic retionapthy classification. In 2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), Coimbatore, India, pp. 1-4. <https://doi.org/10.1109/ICCIC.2017.8524342>
- [10] Roy, A., Dutta, D., Bhattacharya, P., Choudhury, S. (2017). Filter and fuzzy c means based feature extraction and classification of diabetic retinopathy using support vector machines. In 2017 International Conference on Communication and Signal Processing (ICCSP), Chennai, India, pp. 1844-1848. <https://doi.org/10.1109/ICCSP.2017.8286715>
- [11] Chetoui, M., Akhloufi, M.A. (2020). Explainable diabetic retinopathy using EfficientNET. In 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Montreal, QC, Canada, pp. 1966-1969. <https://doi.org/10.1109/EMBC44109.2020.9175664>
- [12] Mane, D.T., Sangve, S.M., Kumbharkar, P.B., Ratnaparkhi, S., Upadhye, G., Borde, S. (2023). A diabetic retinopathy detection using customized convolutional neural network. *International Journal of Electrical and Electronics Research*, 11(2): 609-615. <https://doi.org/10.37391/ijeer.110250>
- [13] Yu, S., Xiao, D., Kanagasingam, Y. (2017). Exudate detection for diabetic retinopathy with convolutional neural networks. In 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Jeju, Korea (South), pp. 1744-1747. <https://doi.org/10.1109/EMBC.2017.8037180>
- [14] Jadhav, A.S., Patil, P.B., Biradar, S. (2021). Optimal feature selection-based diabetic retinopathy detection using improved rider optimization algorithm enabled with deep learning. *Evolutionary Intelligence*, 14: 1431-1448. <https://doi.org/10.1007/s12065-020-00400-0>
- [15] Suryawanshi, V., Setpal, S. (2017). Gaussian transformed GLCM features for classifying diabetic retinopathy. In 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai, India, pp. 1108-1111. <https://doi.org/10.1109/ICECDS.2017.8389612>
- [16] Kumar, S., Kumar, B. (2018). Diabetic retinopathy detection by extracting area and number of microaneurysm from colour fundus image. In 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, pp. 359-364. <https://doi.org/10.1109/SPIN.2018.8474264>
- [17] Alzami, F., Megantara, R.A., Fanani, A.Z. (2019). Diabetic retinopathy grade classification based on fractal analysis and random forest. In 2019 International Seminar on Application for Technology of Information and Communication (iSemantic), Semarang, Indonesia, pp. 272-276. <https://doi.org/10.1109/ISEMANTIC.2019.8884217>
- [18] Carrera, E.V., González, A., Carrera, R. (2017). Automated detection of diabetic retinopathy using SVM. In 2017 IEEE XXIV International Conference on Electronics, Electrical Engineering and Computing (INTERCON), Cusco, Peru, pp. 1-4. <https://doi.org/10.1109/INTERCON.2017.8079692>
- [19] Bilal, A., Zhu, L., Deng, A., Lu, H., Wu, N. (2022). AI-based automatic detection and classification of diabetic retinopathy using U-Net and deep learning. *Symmetry*, 14(7): 1427. <https://doi.org/10.3390/sym14071427>
- [20] Jabbar, M.K., Yan, J., Xu, H., Ur Rehman, Z., Jabbar, A. (2022). Transfer learning-based model for diabetic retinopathy diagnosis using retinal images. *Brain Sciences*, 12(5): 535. <https://doi.org/10.3390/brainsci12050535>