Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 15, Issue. 1 : 2024
ISSN : **1906-9685**

MATLAB - BASED IMAGE PROCESSING SOLUTIONS FOR ENHANCED DIABETIC RETINOPATHY DETECTION

G. Sree Lakshmi 1*, K. Deekshitha[1], B. Sri Vidya [1], B. Sahithi[1], G. Sudeepthi[1] 1*Department of Electronics & Communication Engineering, NRI Institute of Technology, Pothavarappadu (V), Agiripalli (M), Eluru (Dt)-521212

Abstract:
Diabetic retinopathy (DR) is a serious disease which affects vision of diabetic patients. This may cause partial and complete blindness. There are so many works have been carried in the field of Diabetic Retinopathy diagnosis but still lacking some where. Proposes system uses the hybrid technique to build the system effectively and performance has been enhanced in all manners. The proposed system is based on Sobel Edge Detection and Color mapping techniques; where fundus images are processed constructively and obtain better level of accuracy. Fundus image contains exudates, and the principal task of color mapping is to transfer the color to another pigment that enhances the visibility of exudates. The objective of color mapping is to calibrate the color and adjust it accordingly as per the visibility of background and foreground. There is a bit difference between the color of exudates and normal cells or blood vessels. Color mapping classifies the exudates and blood vessels; that definitely contribute to extract the features more precisely. System uses Sobel Edge Detection for extracting the horizontal and vertical edges. It calculates the gradient of fundus image intensities at each pixel. It can extract the shape of exudates, and density can be computed; that later compare with the threshold value.
Keywords - Diabetic Retinopathy, Fundus Image, Sobel Edge Detection, Color Mapping, Exudates Extraction, Retina, Optic Cup

1.Introduction

Diabetic Retinopathy (DR) is a common complication of diabetes mellitus, which causes lesions on the retina that effect vision. If it is not detected early, it can lead to blindness. Unfortunately, DR is not a reversible process, and treatment only sustains vision. DR early detection and treatment can significantly reduce the risk of vision loss. The manual diagnosis process of DR retina fundus images by ophthalmologists is time-, effort-, and cost-consuming and prone to misdiagnosis unlike computer-aided diagnosis systems. Recently, deep learning has become one of the most common techniques that has achieved better performance in many areas, especially in medical image analysis and classification. Convolutional neural networks are more widely used as a deep learning method in medical image analysis and they are highly effective. For this article, the recent state-of-the-art methods of DR color fundus images detection and classification using deep learning techniques have been reviewed and analyzed. Furthermore, the DR available datasets for the color fundus retina have been reviewed. Difference challenging issues that require more investigation are also discussed. Millions of people suffer from Diabetic retinopathy, the leading cause of blindness among working aged adults. Aravind Eye Hospital in India hopes to detect and prevent this disease among people living in rural areas where medical screening is difficult to conduct. Currently, the technicians travel to these rural areas to capture images and then rely on highly trained doctors to review the images and provide diagnosis. Diabetic Retinopathy (DR) is the most common cause of avoidable vision loss, predominantly affecting the working-age population across the globe. Screening for DR, coupled with timely consultation and treatment, is a globally trusted policy to avoid vision loss. However, implementation of DR screening programs is challenging due to the scarcity of medical professionals able to screen a growing global diabetic population at risk for DR. Computer-aided disease diagnosis in retinal image analysis could provide a sustainable approach for such large-scale screening effort. The recent scientific advances in computing capacity and machine learning approaches provide an avenue for biomedical scientists to reach this goal. Aiming to advance the state-of-the-art in automatic DR diagnosis, a grand challenge on "Diabetic Retinopathy – Segmentation and Grading" was organized in conjunction with the IEEE International Symposium on Biomedical Imaging (ISBI - 2018). In this paper, we report the set-up and results of this challenge that is primarily based on Indian Diabetic Retinopathy Image Dataset (IDRiD). There were three principal sub- challenges: lesion segmentation, disease severity grading, and localization of retinal landmarks and segmentation. These multiple tasks in this challenge allow to test the generalizability of algorithms, and this is what makes it different from existing ones. It received a positive response from the scientific community with 148 submissions from 495 registrations effectively entered in this challenge. This paper outlines the challenge, its organization, the dataset used, evaluation methods and results of top-performing participating solutions. The top-performing approaches utilized a blend of clinical information, data augmentation, and an ensemble of models. These findings have the potential to enable new developments in retinal image analysis and image-based DR screening in particular. Automated detection of lesions in retinal images can assist in early diagnosis and screening of a common disease: Diabetic Retinopathy. A robust and computationally efficient approach for the localization of the different features and lesions in a fundus retinal image is presented in this paper. Since many features have common intensity properties, geometric features and correlations are used to distinguish between them. We propose a new constraint for optic disk detection where we first detect the major blood vessels and use the intersection of these to find the approximate location of the optic disk. This is further localized using color properties. We also show that many of the features such as the blood vessels, exudates and microaneurysms and hemorrhages can be detected quite accurately using different morphological operations applied appropriately.

2.Proposed System

The proposed system leverages Color mapping and Sobel edge detection to achieve an innovative approach for diabetic retinopathy lesion detection in fundus images. The Sobel operator is applied for gradient-based edge detection, and the is employed to enhance the completeness and visibility of the segmented edges. This system offers advantages such as enhanced lesion localization, improved lesion completeness, increased contrast, reduced false negatives, and compatibility with color mapping. The applications range from early diabetic retinopathy diagnosis to disease progression monitoring, automated screening programs, telemedicine integration, and clinical decision support for ophthalmologists. The synergy of these techniques in the proposed system aims to significantly improve the accuracy and efficiency of diabetic retinopathy detection.

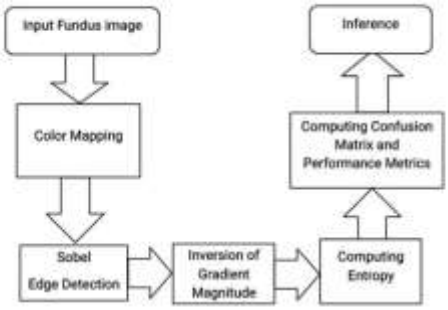


Figure.1.BLOCK DIAGRAM

3.2.1 DATASET

Dataset Overview:

The Kaggle Diabetic Retinopathy Detection challenge dataset, known as Kaggle- EyePACS, comprises a diverse collection of high-resolution RGB retina images captured under various imaging conditions. These images exhibit differing resolutions, ranging from low to high, offering a comprehensive representation of diabetic retinopathy cases. The dataset is divided into a training set and a test set, containing a substantial number of images for algorithmic training and evaluation. A clinician has meticulously graded each image according to the International Clinical Diabetic Retinopathy (ICDR) scale, providing valuable annotations for model development.

1. Image Samples:
• The datasets usually contain a large number of fundus images of the retina. These images are taken using specialized cameras and may vary in terms of resolution, color, and quality.

2. Labels:
• Each image is associated with one or more labels indicating the presence and severity of diabetic retinopathy. Labels may include categories such as

"No Diabetic Retinopathy," "Mild," "Moderate," "Severe," and "Proliferative."

3. Train and Test Sets:
• Datasets are often divided into training and testing sets. The training set is used to train machine learning models, while the test set is used to evaluate the model's performance on new, unseen data.

4. Annotations:
• Some datasets may include additional annotations, such as the location of specific lesions or abnormalities within the images. This information can be valuable for research and algorithm development.

5. Metadata:
• Datasets may include additional metadata, such as patient information, image acquisition details, and other relevant information.

When working with these datasets, researchers and data scientists often engage in tasks like image classification or object detection. They use machine learning techniques to develop models that can automatically detect and classify the severity of diabetic retinopathy in fundus images.

Considerations for Analysis:

A notable portion of the Kaggle-EyePACS images is characterized as ungradable due to factors like loss of focus, under or overexposure. Graders, following a suggested procedure, manually evaluated the quality of all images, resulting in a refined dataset. The study explores the implications of gradability estimates on algorithmic performance compared to utilizing the full dataset. The authors emphasize the potential role of non- experts in performing image quality estimation, raising questions about the automation of this preprocessing step. In subsequent experiments, the study investigates the consequences of incorporating gradability estimates, providing insights into the robustness and generalizability of diabetic retinopathy detection models trained on Kaggle-EyePACS .

COLOR MAPPING

Colour mapping, also known as colour transfer or colour grading, is a technique used in image processing to change the colours of an image to achieve a specific visual effect. This process is employed for various purposes, including enhancing the aesthetic appeal of images, improving visibility, or adapting images for specific applications. Colour mapping can be applied for artistic reasons, to correct colour imbalances, or to visualize data in a more understandable way.

Here are key aspects of colour mapping in image processing:

1. Colour Models:
• Colour mapping often involves working with different colour models, such as RGB (Red, Green, Blue), HSV (Hue, Saturation, Value), or LAB. These models represent colours in different ways, and conversions between them may be necessary during colour mapping.

2. Colour Transfer Functions:
• Colour mapping is typically achieved through the use of transfer functions. A transfer function maps the input colours of an image to new colours based on a mathematical transformation. This function defines how each colour component (e.g., red, green, blue) is modified.

3. Colour Grading:

- Colour mapping is extensively used in colour grading for images and videos. In colour grading, the aim is often to achieve a particular mood, atmosphere, or stylistic effect. This process involves adjusting the colour balance, contrast, and saturation to create a desired look.

4. Histogram Equalization:

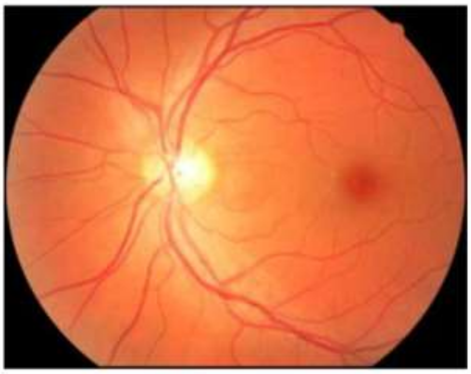
- Histogram equalization is a technique used in colour mapping to enhance the contrast of an image by spreading out the intensity values. It is particularly useful when an image has low contrast or when certain colour ranges dominate the image.

Understanding and mastering colour mapping techniques empower image processing professionals and artists to manipulate visual content creatively and effectively. The choice of colour mapping strategy depends on the specific requirements of the task and the intended visual impact on the audience. As technology and algorithms evolve, colour mapping continues to play a crucial role in shaping the aesthetics and usability of digital imagery.

RGB to GRAY:

An RGB image can be viewed as three images (a red scale image, a green scale image and a blue scale image) stacked on top of each other. In MATLAB, an RGB image is basically a M*N*3 array of color pixel, where each color pixel is a triplet which corresponds to red, blue and green color component of RGB image at a specified spatial location. Similarly, A Grayscale image can be viewed as a single layered image. In MATLAB, a grayscale image is basically M*N array whose values have been scaled to represent intensities. In MATLAB, there is a function called rgb2gray() is available to convert RGB image to grayscale image. Here we will convert an RGB image to grayscale image without using rgb2gray() function. Our key idea is to convert an RGB image pixel which a triplet value corresponding to red, blue and green colour component of an image at a specified spatial location to a single value by calculating a weighted sum of all three color component.

We convert an image from RGB to grayscale during image processing because it simplifies the image and reduces the amount of information needed to represent it. Grayscale images only have shades of gray, which makes them easier to analyze and process compared to color images. It can also help in reducing computational complexity for certain algorithms.



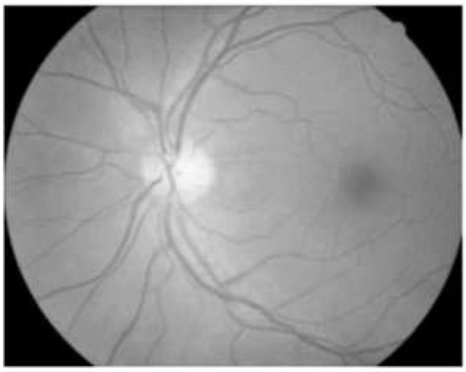


Figure.2. NORMAL FUNDUS IMAGE Figure.3. GRAYSCALED IMAGE

Algorithm for conversion:

1.

Read RGB color image into MATLAB environment

2.Extract Red, blue and green colour components from RGB image into 3 different 2-D matrices

3.Create a new matrix with the same number of rows and columns as RGB image, containing all zeros

4.Convert each RGB pixel values at location (i, j) to grayscale values by forming a weighted sum of the Red, Green, and Blue colour components and assign it to corresponding location (i, j) in new matrix

5.grayscale value at (i, j) = 0.2989 * R(i, j) + 0.5870 * G(i, j) + 0.1140 * B(i, j)

JET COLORMAPPING:

Jet conversion in image processing typically refers to the process of converting a scalar field into a color representation using a predefined color map, often referred to as a "jet colormap." The jet colormap is a popular choice for visualizing scalar data, such as elevation maps, heat maps, or any other data that can be represented by a single scalar value at each pixel.

The jet colormap assigns different colors to different ranges of scalar values. Typically, lower values are represented by blue or green colors, intermediate values by yellow or orange, and higher values by red or magenta. This color mapping helps to visually highlight variations in the scalar field.The jet conversion process involves associating each scalar value in the image with a corresponding color based on the defined colormap. This color mapping can aid in the interpretation and visualization of data, making it easier to identify trends, patterns, or anomalies within the image.

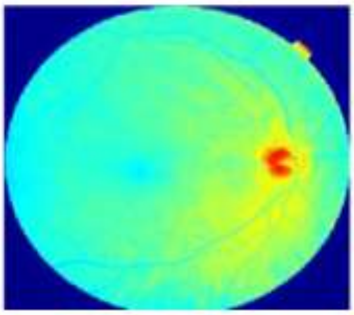


Figure.3.COLOR MAPPED FUNDUS IMAGE

SOBEL EDGE DETECTION

Methods of Edge Detection

There are various methods, and the following are some of the most commonly usedmethods-

- Prewitt edge detection
- Sobel edge detection
- Laplacian edge detection
- Canny edge detection

SOBEL EDGE DETECTION:

Sobel edge detection is a popular method in image processing and computer visionfor detecting edges in an image. It belongs to a category of edge detection algorithms thataim to identify boundaries within the image where there are significant changes in intensityor color.The Sobel operator uses convolution with Sobel kernels to compute the gradient ofthe image intensity in both the horizontal and vertical directions. The gradients represent the rate of change of intensity at each pixel, and high gradient values are indicative of an edge.

The Sobel kernels are small matrices that are convolved with the image. For edge detection, two kernels are commonly used:

1. Sobel Horizontal Kernel (Gx):

$$Gx = \begin{matrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{matrix}$$

2. Sobel Vertical Kernel (Gy):

$$Gy = \begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix}$$

Convolution:

Convolve the image with the horizontal Sobel kernel to obtain the gradient in the horizontal direction (Gx). Then, convolve the image with the vertical Sobel kernel to obtain the gradient in the vertical direction (Gy).The convolution operation involves sliding the Sobel kernel over the image,computing the sum of products at each position.

Gx = Image * Horizontal Sobel Kernel Gy = Image * Vertical Sobel Kernel

Gradient Magnitude:

The magnitude of the gradient indicates the strength of the edge, and the directiongives the orientation of the edge. Once the gradients are computed, an edge strength threshold can be applied to identify pixels with significant changes in intensity, highlighting the edges in the image. Combine the horizontal and vertical gradients to calculate the gradient magnitudeusing the following formula:

Gradient Magnitude (G) = sqrt(Gx^2 + Gy^2)

This computes the magnitude of the gradient at each pixel, indicating the strength of theedge.

Thresholding:

Apply a threshold to the gradient magnitude. Pixels with a gradient magnitudeabove a certain threshold are considered part of an edge, while others are suppressed. This step helps eliminate noise and retain only significant edges.

If G > Threshold, Edge PixelElse, Non-Edge Pixel

Sobel edge detection is simple and computationally efficient, making it a widely used technique for preprocessing steps in various computer vision applications. However,it may be sensitive to noise, and more advanced edge detection methods, such as the Cannyedge detector, are often employed for improved performance in certain scenarios.

Normalization of Gradient Magnitude:

Normalization process is a common practice in image processing to bring gradientmagnitudes to a consistent and standardized scale. By ensuring that the values fall within the[0, 1] range, it facilitates visual interpretation, analysis, or comparison of images, particularly in scenarios where consistent intensity scaling is crucial.

The first step involves determining the maximum value within the entire ‘gradientMagnitude’ matrix. This is achieved using the max function, which identifies themaximum value when the matrix is linearized into a single column. The result is stored in a variable called ‘maxValue’.

Subsequently, the actual normalization takes place. Each element in the original ‘gradientMagnitude’ matrix is divided by the previously computed ‘maxValue’. This division operation scales the entire range of gradient magnitudes so that the maximum value becomes 1.

INVERTING GRADIET MAGNITUDE

Noise Removal using Gaussian Smoothing:

In the initial step, the code addresses the presence of noise in the gradient magnitude image. The Gaussian smoothing technique is applied, utilizing the `imgaussfilt` function. The parameter `sigma` is introduced to control the standard deviation of the Gaussian filter. This step is crucial for enhancing the quality of the gradient magnitude image by reducing unwanted noise, ensuring a cleaner representation of the underlying features.

Background Removal using Morphological Operation:

Following noise reduction, the focus shifts to background removal through morphological operations. The top-hat transformation, facilitated by the `imtophat` function, is employed for this purpose. A disk-shaped structuring element (`se`) is defined, acting as a filter to selectively remove background features. The size of this structuring element is adaptable, allowing adjustment based on the specific characteristics of the background in the image. The resulting image after this step reflects a reduction in background influence, emphasizing the gradient-related information.

Visualization:

To assess the effectiveness of the noise and background removal processes, the code includes a visualization step. The `imshow` function is employed to display the processed image. Additionally, a "jet" colormap is chosen for visualization purposes, enhancing the differentiation of features in the displayed image. The addition of a title, "Final Image," aids in conveying the completion of the processing steps and provides context to the viewer.

Adjustment Parameters:

The `sigma` parameter in Gaussian smoothing is adjustable, allowing users to fine- tune noise reduction based on the observed level of noise in the gradient magnitude image.

The disk size of the structuring element (`se`) is modifiable, enabling users to tailor background removal according to the size of background features in the image.

ENTROPY

In information theory and image processing, entropy is a measure of the randomness or uncertainty in a set of data. In the context of images, entropy quantifies the amount of information or complexity present in the distribution of pixel intensities. High entropy values indicate greater unpredictability or complexity, while low entropy values suggest a more regular or ordered distribution of pixel values.
The formula for entropy is given by:
H(X) is the entropy of the data.
P(xi) is the probability mass (probability of occurrence) of the data point xi .
In the context of image processing, the entropy of an image is often calculated based on the histogram of pixel intensities. The entropy formula is adapted to:
where:
p(i) is the probability of occurrence of pixel intensity i in the image. N is the total number of pixels in the image.
L is the number of possible intensity levels (typically 256 for an 8-bit grayscale image).
The entropy is a measure that reflects the amount of information or disorder in an image's pixel intensity distribution. It finds applications in various fields, including image processing, data compression, and information theory.
EVALUATING CONFUSION MATRIX AND PERFORMANCE METRICS:
Binary Conversion:

Binary conversion is the initial step where the original image 'img' is transformed into binary images 'gt' (ground truth) and 'si' (segmented image). This conversion is crucialfor creating binary masks representing the presence or absence of features, facilitating thecomparison between the ground truth and the segmented result.
Confusion Matrix Calculation:
The confusion matrix is a powerful tool for evaluating the performance of a binaryclassification or segmentation system. It summarizes the classification results by countingoccurrences of true positive (TP), true negative (TN), false positive (FP), and false negative(FN). The confusionmat function is employed to compute this matrix using the binary representations of the ground truth and segmented images.
Need for Confusion Matrix:
Quantitative Evaluation:
The confusion matrix provides a quantitative assessment of the performance of a binary segmentation algorithm. It allows for a detailed breakdown of correct and incorrectclassifications.
Sensitivity and Specificity:
TP, TN, FP, and FN are essential for computing metrics such as sensitivity (recall),specificity, precision, and accuracy. These metrics offer insights into the algorithm's abilityto correctly identify positive and negative instances.
Error Analysis:
Understanding the distribution of pixels across the matrix helps in error analysis. False positives and false negatives can be identified, allowing for targeted improvements to the segmentation algorithm.
Optimization:
The confusion matrix aids in optimizing the performance of the segmentation algorithm by revealing areas of strength and weakness. Adjustments to the algorithm can be made based on the observed classification results.
Extracting Values from Confusion Matrix:
Extracting individual elements from the confusion matrix is the next step. These values represent the count of pixels in each category: TP, TN, FP, and FN.
True Positives (TP):
TP represents the count of pixels that are correctly classified as positive (e.g., correctly identified objects or features) in both the ground truth and segmented images. Inthe confusion matrix, true positives are located at the intersection of the positive class (1) in both ground truth and segmented images.
True Negatives (TN):

TN is the count of pixels correctly classified as negative (e.g., correctly identified background or absence of features) in both the ground truth and segmented images. In theconfusion matrix, true negatives are located at the intersection of the negative class (0) in both ground truth and segmented images.
False Positives (FP):
FP represents the count of pixels incorrectly classified as positive in the segmentedimage when they are actually negative in the ground truth. False positives are located in the row corresponding to the negative class in the ground truth and the column corresponding to the positive class in the segmented image.
False Negatives (FN):
FN is the count of pixels incorrectly classified as negative in the segmented imagewhen they are actually positive in the ground truth. False negatives are located in the row corresponding to the positive class in the ground truth and the column corresponding to thenegative class in the segmented image.
Evaluation Metrics for Binary Image Segmentation:
Specificity:
Definition:
Specificity measures the ability of the classifier to correctly identify the negative class. It is calculated as the ratio of true negatives (TN) to the sum of true negatives and false positives (FP).
Calculation:
specificity = TN / (TN + FP)Interpretation:
A high specificity indicates a low rate of false positives, meaning the classifier iseffective at correctly identifying the absence of features.
Precision:
Definition:
Precision represents the accuracy of the positive predictions made by the classifier.It is calculated as the ratio of true positives (TP) to the sum of true positives and false positives (FP).
Calculation:
precision = TP / (TP + FP)Interpretation:
High precision indicates a low rate of false positives, meaning that when theclassifier predicts positive, it is accurate.
Accuracy:
Definition:
Accuracy measures the overall correctness of the classifier and is calculated as theratio of the sum of true positives and true negatives to the total number of instances.
Calculation:
accuracy = (TP + TN) / (TP + TN + FP + FN)
Interpretation:
Accuracy provides an overall measure of the classifier's correctness across bothpositive and negative predictions.
Recall:
Definition:
Recall, also known as sensitivity, measures the ability of the classifier to correctlyidentify the positive class. It is calculated as the ratio of true positives (TP) to the sum of true positives and false negatives (FN).
Calculation:
recall = TP / (TP + FN)Interpretation:
High recall indicates a low rate of false negatives, meaning the classifier is effectiveat correctly identifying positive instances.
F1 Score:
Definition:
The F1 score is the harmonic mean of precision and recall, providing a balancedmeasure of the classifier's performance.

Calculation:
f1_score = 2 * (precision * recall) / (precision + recall);Interpretation:
F1 score considers both false positives and false negatives, providing acomprehensive metric that balances precision and recall.

3.Results and Discussion

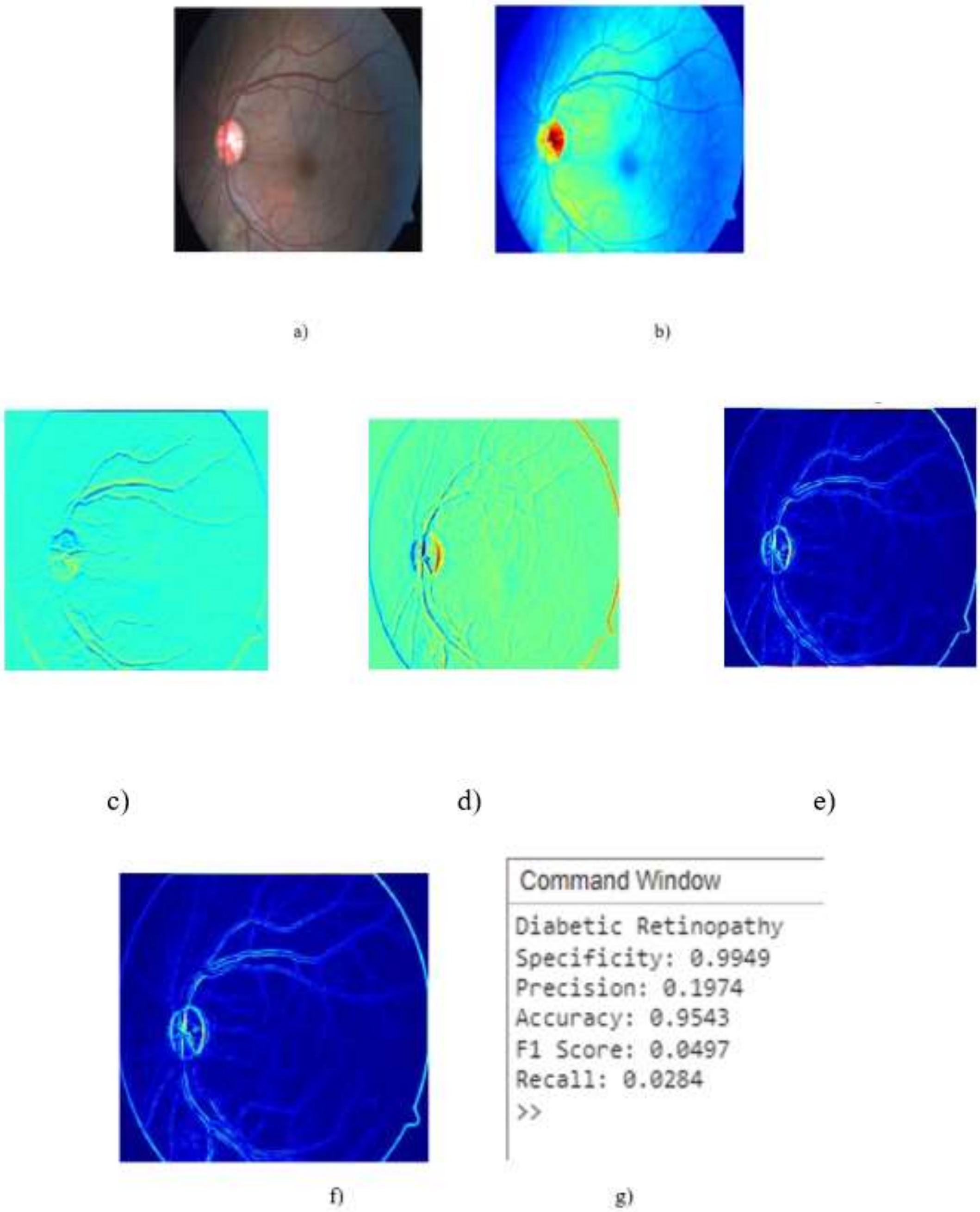


Figure.4. a)INPUT FUNDUS IMAGE b)COLOR MAPPING OF INPUT FUNDUS IMAGE c)HORIZONTAL GRADIENT d) VERTICAL GRADIENTe) GRADIENT MAGNITUDE f)FINAL IMAGE g) OUTPUT AT COMMAND WINDOW

Here, the input fundus image is given to the algorithm and the result is displayed as patient is suffering from Diabetic Retinopathy.

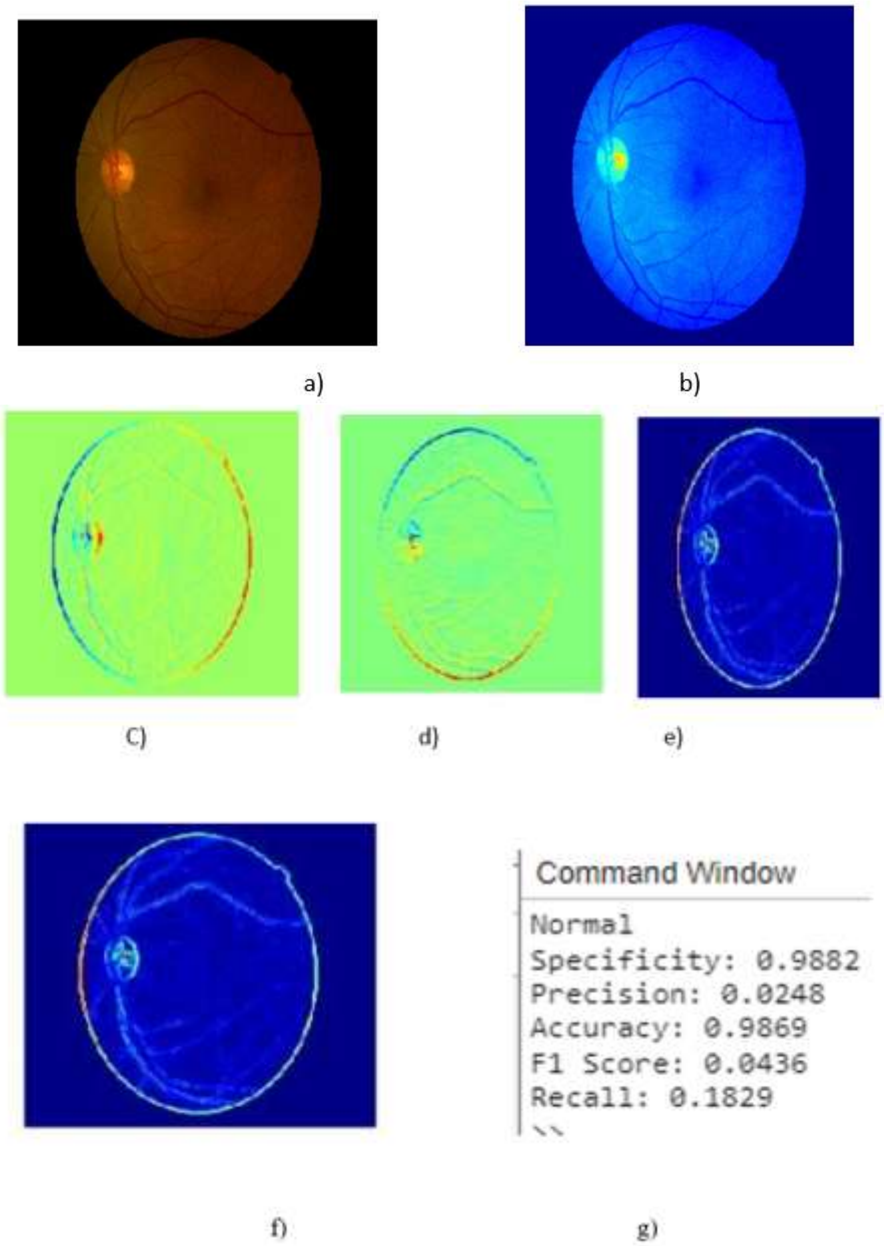


Figure.5. a)INPUT FUNDUS IMAGE b)COLOR MAPPING OF INPUT FUNDUS IMAGE c)HORIZONTALGRADIENT d) VERTICAL GRADIENT e) GRADIENT MAGNITUDE f) FINAL IMAGE g) OUTPUT AT COMMAND WINDOW

Here, the input fundus image is given to the algorithm and the result is displayed as patient has Normal eye (not suffering from Diabetic Retinopathy).

4.Conclusion

In conclusion, the Diabetic Retinopathy detection project employs advanced image processing techniques, including Sobel edge detection and morphological operations, to enhance lesion localization and completeness in fundus images. The algorithm's adaptability to color mapping, compatibility with binary image processing, and comprehensive performance metrics contribute to its robustness and reliability. With applications ranging from early diagnosis to telemedicine and research, the project showcases its versatility and potential impact on diabetic retinopathy management. The algorithm's integration into healthcare systems and its ability to contribute to population health studies underscore its significance in advancing ophthalmic healthcare, paving the way for improved diagnostic accuracy and timely interventions in the detection of diabetic retinopathy.

References

[1] Wejdan L. Ayoubi, Wafaa M. Shalash, Maysoon F. Abulkhair, Diabetic retinopathy detection through deep learning techniques: A review, Informatics in Medicine Unlocked, Volume 20, 2020.

[2] Kaggle, APTOS : Eye Preprocessing in Diabetic Retinopathy, https://www.kaggle.com/code/ratthachat/aptos-eye-preprocessing-indiabetic-retinopathy/notebook. Accessed 07 April 2022.

[3] Towards Data Science, Blindness detection (Diabetic retinopathy) using Deep learning on Eye retina images, https://towardsdatascience.com/blindness-detection- diabeticretinopathy-using-deep-learning-on-eye-retina-images-baf20fcf409e. Accessed 07 April 2022.

[4] Prasanna Porwal, Samiksha Pachhade, et al. IDRiD: Diabetic Retinopathy – Segmentation and Grading Challenge, Medical Image Analysis, Volume 59, 2020.

[5] S. Ravishankar, A. Jain and A. Mittal, "Automated feature extraction for early detection of diabetic retinopathy in fundus images," 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, 2009, pp. 210-217.

[6] K. K. Palavalasa and B. Sambaturu, "Automatic Diabetic Retinopathy Detection Using Digital Image Processing," 2018 International Conference on Communication and Signal Processing (ICCSP), Chennai, 2018, pp. 0072-0076, doi: 10.1109/ICCSP.2018.8524234.

[7] A. Osareh, B. Shadgar, and R. Markham, "A computation a intelligence-based approach for detection of exudates in diabetic retinopathy images," IEEE Trans. Inf. Technol.
Biomed., vol. 13, no. 4, pp. 535–545, 2009

[8] Gardner, G & Keating, David & Williamson, Tom & Elliott, A. (1996). Automatic detection of diabetic retinopathy using an artificial neural network: A screening tool. The British journal of ophthalmology. 80. 940-4. 10.1136/bjo.80.11.940.

[9] M. M. Dharmana and A. M.S., "Pre-diagnosis of Diabetic Retinopathy using Blob Detection," 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), 2020, pp. 98-101, Doi: 10.1109/ICIRCA48905.2020.9183241.

[10] M. Arora and M. Pandey, "Deep Neural Network for Diabetic Retinopathy Detection," 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (ComiCon), 2019, pp. 189-193, Doi: 10.1109/COMITCon.2019.8862217.

[11] Y. S. Boral and S. S. Thorat, "Classification of Diabetic Retinopathy based on Hybrid Neural Network," 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), 2021, pp. 1354-1358, Doi: 10.1109/ICCMC51019.2021.9418224. [12] Kalyani, G., Janaki Ramaiah, B., Karuna, A. et al. Diabetic retinopathy detection and classification using capsule networks. Complex Intell. Syst. (2021). https://doi.org/10.1007/s40747-021-00318-9

[13] Jyoti Patil , Dr A. L. Chaudhari, Detection of Diabetic Retinopathy Using Sobel edge detection method in DIP, International Journal of Scientific & Engineering Research Volume 3, Issue 7, July-2012. [14] Kanimozhi, J., Vasuki, P. & Roomi, S.M.M. Fundus image lesion detection algorithm for diabetic retinopathy screening. J Ambient Intell Human Compute 12, 7407–7416 (2021).
https://doi.org/10.1007/s12652-020-02417-w

INVERTING GRADIET MAGNITUDE

Noise Removal using Gaussian Smoothing:

In the initial step, the code addresses the presence of noise in the gradient magnitude image. The Gaussian smoothing technique is applied, utilizing the `imgaussfilt`function. The parameter `sigma` is introduced to control the standard deviation of the Gaussian filter. This step is crucial for enhancing the quality of the gradient magnitude image by reducing unwanted noise, ensuring a cleaner representation of the underlying features.

Background Removal using Morphological Operation:

Following noise reduction, the focus shifts to background removal through morphological operations. The top-hat transformation, facilitated by the `imtophat` function, is employed for this purpose. A disk-shaped structuring element (`se`) is defined,acting as a filter to selectively remove background features. The size of this structuring element is adaptable, allowing adjustment based on the specific characteristics of the background in the image. The resulting image after this step reflects a reduction in background influence, emphasizing the gradient-related information.

Visualization:

To assess the effectiveness of the noise and background removal processes, the code includes a visualization step. The `imshow` function is employed to display the processed image. Additionally, a "jet" colormap is chosen for visualization purposes, enhancing the differentiation of features in the displayed image. The addition of a title, "Final Image," aids in conveying the completion of the processing steps and provides context to the viewer.

Adjustment Parameters:

The `sigma` parameter in Gaussian smoothing is adjustable, allowing users to fine-tune noise reduction based on the observed level of noise in the gradient magnitude image.

The disk size of the structuring element (`se`) is modifiable, enabling users to tailorbackground removal according to the size of background features in the image.

ENTROPY

In information theory and image processing, entropy is a measure of the randomness or uncertainty in a set of data. In the context of images, entropy quantifies theamount of information or complexity present in the distribution of pixel intensities. High entropy values indicate greater unpredictability or complexity, while low entropy values suggest a more regular or ordered distribution of pixel values.

The formula for entropy is given by:

$$H(X) = -\sum_i P(x_i) \log_2(P(x_i))$$

H(X) is the entropy of the data.

P(xi) is the probability mass (probability of occurrence) of the data point *xi* .

In the context of image processing, the entropy of an image is often calculated basedon the histogram of pixel intensities. The entropy formula is adapted to:

$$H(X) = -\sum_{i=1}^{256} \left(\frac{p(i)}{N}\right) \log_2 \left(\frac{p(i)}{N}\right)$$

where:

p(i) is the probability of occurrence of pixel intensity i in the image.N is the total number of pixels in the image.

L is the number of possible intensity levels (typically 256 for an 8-bit grayscale image).

The entropy is a measure that reflects the amount of information or disorder in an image's pixel intensity distribution. It finds applications in various fields, including image processing, data compression, and information theory.

3.1.1 EVALUATING CONFUSION MATRIX AND PERFORMANCE METRICS:

Binary Conversion:

Binary conversion is the initial step where the original image 'img' is transformed into binary images 'gt' (ground truth) and 'si' (segmented image). This conversion is crucialfor creating binary masks representing the presence or absence of features, facilitating thecomparison between the ground truth and the segmented result.

Confusion Matrix Calculation:

The confusion matrix is a powerful tool for evaluating the performance of a binaryclassification or segmentation system. It summarizes the classification results by countingoccurrences of true positive (TP), true negative (TN), false positive (FP), and false negative(FN). The confusionmat function is employed to compute this matrix using the binary representations of the ground truth and segmented images.

Need for Confusion Matrix:

Quantitative Evaluation:

The confusion matrix provides a quantitative assessment of the performance of a binary segmentation algorithm. It allows for a detailed breakdown of correct and incorrectclassifications.

Sensitivity and Specificity:

TP, TN, FP, and FN are essential for computing metrics such as sensitivity (recall),specificity, precision, and accuracy. These metrics offer insights into the algorithm's abilityto correctly identify positive and negative instances.

Error Analysis:

Understanding the distribution of pixels across the matrix helps in error analysis. False positives and false negatives can be identified, allowing for targeted improvements to the segmentation algorithm.

Optimization:

The confusion matrix aids in optimizing the performance of the segmentation algorithm by revealing areas of strength and weakness. Adjustments to the algorithm can be made based on the observed classification results.

Extracting Values from Confusion Matrix:

Extracting individual elements from the confusion matrix is the next step. These values represent the count of pixels in each category: TP, TN, FP, and FN.

True Positives (TP):

TP represents the count of pixels that are correctly classified as positive (e.g., correctly identified objects or features) in both the ground truth and segmented images. Inthe confusion matrix, true positives are located at the intersection of the positive class (1) in both ground truth and segmented images.

True Negatives (TN):

TN is the count of pixels correctly classified as negative (e.g., correctly identified background or absence of features) in both the ground truth and segmented images. In theconfusion matrix, true negatives are located at the intersection of the negative class (0) in both ground truth and segmented images.

False Positives (FP):

FP represents the count of pixels incorrectly classified as positive in the segmentedimage when they are actually negative in the ground truth. False positives are located in the row corresponding to the negative class in the ground truth and the column corresponding to the positive class in the segmented image.

False Negatives (FN):

FN is the count of pixels incorrectly classified as negative in the segmented imagewhen they are actually positive in the ground truth. False negatives are located in the row corresponding to the positive class in the ground truth and the column corresponding to thenegative class in the segmented image.

Evaluation Metrics for Binary Image Segmentation:

1. Specificity:

Definition:

Specificity measures the ability of the classifier to correctly identify the negative class. It is calculated as the ratio of true negatives (TN) to the sum of true negatives and false positives (FP).

Calculation:

specificity = TN / (TN + FP)Interpretation:

A high specificity indicates a low rate of false positives, meaning the classifier iseffective at correctly identifying the absence of features.

Precision:

Definition:

Precision represents the accuracy of the positive predictions made by the classifier. It is calculated as the ratio of true positives (TP) to the sum of true positives and false positives (FP).

Calculation:

precision = TP / (TP + FP)Interpretation:

High precision indicates a low rate of false positives, meaning that when theclassifier predicts positive, it is accurate.

Accuracy:

Definition:

Accuracy measures the overall correctness of the classifier and is calculated as theratio of the sum of true positives and true negatives to the total number of instances

.

Calculation:

accuracy = (TP + TN) / (TP + TN + FP + FN)

Interpretation:

Accuracy provides an overall measure of the classifier's correctness across both positive and negative predictions.

Recall:

Definition:

Recall, also known as sensitivity, measures the ability of the classifier to correctly identify the positive class. It is calculated as the ratio of true positives (TP) to the sum of true positives and false negatives (FN).

Calculation:

recall = TP / (TP + FN)Interpretation:

High recall indicates a low rate of false negatives, meaning the classifier is effectiveat correctly identifying positive instances.

F1 Score:

Definition:

The F1 score is the harmonic mean of precision and recall, providing a balancedmeasure of the classifier's performance.

Calculation:

f1_score = 2 * (precision * recall) / (precision + recall);Interpretation:

F1 score considers both false positives and false negatives, providing acomprehensive metric that balances precision and recall.