



Retinal Vessel Segmentation Using Multi-Scale Residual Convolutional Neural Network (MSR-Net) Combined with Generative Adversarial Networks

Mithun Kumar Kar¹ · Debanga Raj Neog² · Malaya Kumar Nath¹

Received: 11 December 2021 / Revised: 14 September 2022 / Accepted: 15 September 2022 /

Published online: 26 September 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Retinal fundus images provide valuable diagnostic and clinical information in the diagnosis of ophthalmologic diseases. Retinal blood vessel analysis provides important diagnostic information about thinning of the retinal nerve fiber layer and alteration in the structural appearance of the optic nerve head. Here, an accurate retinal vessel detection method is proposed from fundus images using a generative adversarial network (GAN) utilizing multiple loss functions. The proposed GAN architecture consists of the generator as a segmentation network and the discriminator as a classification network. The generator is a multi-scale residual convolutional neural network with skip connection and up-sampling, while the discriminator is a vision transformer that acts as a binary classifier. The inception module extracts multi-scale features of vessel segments from different scales and captures fine vessel segments. The discriminator consists of stacked self-attention networks and position-wise fully connected feed-forward networks inferring two-class output. The attention mechanism in the transformer is competent to preserve both global and local information while acting as a discriminator. The proposed GAN model segments the blood vessels more accurately through the adversarial learning process to produce state-of-the-art results. In the preprocessing stage, the contrast of blood vessels is enhanced by contrast-limited adaptive histogram equalization algorithm. The robustness and efficacy of the proposed

✉ Mithun Kumar Kar
mithunkar.iitg@gmail.com

Debanga Raj Neog
dneog@iitg.ac.in

Malaya Kumar Nath
malaya.nath@gmail.com

¹ Department of Electronics and Communication Engineering, National Institute of Technology Puducherry, Karaikal 609609, India

² Mehta Family School of Data Science and Artificial Intelligence, Indian Institute of Technology Guwahati, Guwahati 781039, India

method have been evaluated on publicly available DRIVE, STARE, CHASE_DB1, HRF, ARIA, IOSTAR, and RC-SLO databases. Different performance measures like accuracy, sensitivity, precision, intersection of union, and F1Score are adopted to compare the proposed method with the existing methods available in the literature. The proposed method attains an accuracy of 0.9873 for CHASE_DB1 database, 0.9742 for DRIVE database, 0.9773 for HRF database, and 0.9628 for ARIA database.

Keywords Blood vessel segmentation · Fundus image · Vision transformer · Inception module · GAN

1 Introduction

Fundus image provides valuable information about the inner structure of eye by projecting different parts like retina, optic disk (OD), fovea, and blood vessels [1]. Retinal blood vessel analysis provides valuable information about thinning of the retinal nerve fiber layer and alteration in the structural appearance of the optic nerve head, which leads to the development of reproducible glaucomatous visual field defects [2]. It is very challenging task to segment retinal blood vessels from the fundus image due to similar color and texture with the background. Many authors have used preprocessing techniques like contrast enhancement and intensity transformations to visualize the blood vessels on the retinal surface.

The vessel segmentation algorithms can be divided into two categories (traditional instruction-based algorithms and automatic machine learning algorithms) [28]. In instruction-based methods, different image processing algorithms like filters, edge detection, morphological, tracking approaches, region-based segmentation, etc., are used, whereas traditional machine learning-based approaches generally used supervised learning algorithms. Till date, the authors used different techniques for automatic and accurate segmentation of blood vessels using various machine learning algorithms. It needs manual efforts for labeling the context used for training the datasets. But, recently developed deep learning algorithms give satisfactory results and eliminate the manual segmentation of vessels. In the last couple of years, a lot of work has been done on vessel segmentation using deep learning techniques. Many authors use different deep neural network (DNN) architectures for vessel segmentation, among which some networks become popular due to their robustness and accuracy. For segmenting biomedical images, UNet [33] is used by the researchers. It is an encoder–decoder architecture, which reveals admirable performance. Recently, an advanced deep learning architecture named generative adversarial network (GAN) is introduced to generate new data from the distribution of known data [16]. It consists of two neural networks, the generator and the discriminator. Both the networks are trained simultaneously. The generator creates new synthetic images or segmentation maps from training database, and the discriminator discriminates human-annotated vessel maps (real label) from machine-generated vessel maps (fake label).

Uysal et al. [38] have implemented fully connected convolutional neural network (CNN) for vessel segmentation from grayscale fundus images. In the preprocessing stage, the authors used gray-level normalization, contrast-limited adaptive histogram

equalization (CLAHE), and gamma correction to make the dataset appropriate for training. The authors have used CNN layer followed by batch normalization and ReLU. Softmax loss function is used for classifying the data into two classes, vessel and non-vessel pixels. Gu et al. [17] have proposed a context encoder network (CENET) to confine more complex high-level information and preserve spatial information for 2D vessel segmentation. A context extractor module (consists of a dense atrous convolution (DAC) block and a residual multi-kernel pooling (RMP) block) is used between encoder and decoder module. These blocks capture additional high-level features and encompass essential spatial information.

Yan et al. [42] have trained the UNet simultaneously with a joint loss, including pixel-wise loss and a segment-level loss. The authors introduced a feature fusion module with a multi-scale convolution block to capture more semantic information. The fusion module preserves the spatial information by combining spatial path with a large kernel. Huazhu and their co-authors [31] proposed a deep vessel architecture consisting of multi-level CNN with side output layers to learn a rich hierarchical representation and model the long-range interactions between pixels by a conditional random field (CRF).

Hu et al. [22] segmented the blood vessels from color fundus images using CNN and fully connected CRFs. It uses multi-scale CNN architecture with an improved cross-entropy loss function to train DRIVE and STARE databases. Shin et al. [36] proposed a vessel graph network (VGN) by linking CNN architecture and graph neural network (GNN). This network jointly models both local appearances and global vessel structures by utilizing semi-regular graph nodes. The authors divided the method into three parts, such as (i) generating pixel-wise features and vessel probabilities, (ii) extracting features to reflect vascular connectivity using GNN, and (iii) inference module to produce the final segmentation map.

Recently, many researchers used GAN to improve the inferences of various tasks, such as synthetic generation and reconstruction [12], image translation [43], image enhancement [34], domain adaption [20], object detection [26], and segmentation [41]. Relating to the segmentation task, different authors used GAN to increase the segmentation performance. Xue et al. [41] proposed a GAN-based segmentation model for brain tumor segmentation, called SegAN. This model consists of an adversarial critic network, and a fully CNN generator trained simultaneously to learn both global and local features for capturing spatial relationships between pixels. Son et al. [37] proposed a model using GAN to generate retinal vessel maps using binary cross-entropy loss.

Guo et al. [18] proposed a neural network architecture based on the Dense UNet using the inception module and GAN for accurate vessel segmentation in the fundus image. They trained both the generator and discriminator alternately using a combined loss function. They have used UNet with dense block and inception module as generator module, whereas the discriminator was a binary classifier using deep neural network. Beom et al. [4] proposed a conditional generative adversarial network called M-GAN for retinal vessel segmentation by balancing losses through stacked deep, fully convolutional networks. The authors have used an M-generator with short-term skip connections and long-term residual connections with a UNet backbone and M-discriminator, a binary classifier with binary cross-entropy (BCE)

loss. The generator consisted of two stacked FCNs with multi-kernel pooling blocks. They implemented multiple loss functions for M-generator and BCE loss function for M-discriminator. These losses are trained alternatively to enhance the output of the generator through adversarial training. Xinghua et al. proposed [27] a deep translation-based change detection network (DTCDN) for optical and SAR images. They used a deep translation network for conversion of optical images to SAR images and a change detection (CD) network for detecting changes between the generated SAR images and the corresponding ground truth images. The translational network consists of no-independent-component-for-encoding GAN (NICE-GAN) network which utilized basic cycle GAN architecture [49] with the introspective networks (INN) in the discriminator part to improve the efficiency of the generator. The change detection network utilizes a UNet++ network [48] incorporating depthwise separable convolution [7] which can produce better segmentation result. They trained the model with weighted multi-scale loss function which significantly reduces the convergence time with catching more information at different scales.

Zhang et al. [46] proposed patch-based deep learning network (Bridge-Net) where both UNet and recurrent neural network (RNN) are used for extracting context information to generate probability maps. The authors have used a patch classification algorithm with a patch-based loss weight mapping to minimize the imbalance between blood vessels and background. Similarly, Xiangyu et al. [9] proposed a deformable convolutional M-shaped network (D-MNet) using multi-scale attention mechanism for blood vessel segmentation. They used a pulse-coupled neural network (PCNN) model along with M-shaped convolutional neural network for multi-threshold segmentation. The D-MNet is capable of extracting multi-angle feature information using convolution kernels of different scales, whereas the multi-scale attention model with residual mechanism changes the acquired feature information into multi-channel information and updates the weight of the feature information of each channel so that the network is capable to distinguish feature information with more accuracy. Danny et al. [6] proposed patches convolution attention-based transformer UNet (PCAT-UNet) network for blood vessel segmentation which is basically an encoder–decoder network comprising patches convolution attention transformer (PCAT) blocks.

The rest part of the paper is organized as follows: Methodology for blood vessel segmentation using the proposed GAN model is discussed in Sect. 2. Experimental results and analysis are discussed for the proposed model in Sect. 3. Ablation study of the proposed model is discussed in Sect. 4. The performance of the proposed method in segmenting low-quality retinal images is discussed in Sect. 5. Also, the computational complexity of the model with two types of discriminator is analyzed in Sect. 6. Finally, the summarization of the work along with future research direction is presented in Sect. 7.

2 Proposed Methodology

This article proposes a new architecture for robust retinal blood vessel segmentation using deep convolutional GAN using multiple losses which can accurately segment the blood vessels w.r.t ground truth. The model consists of preprocessing block along

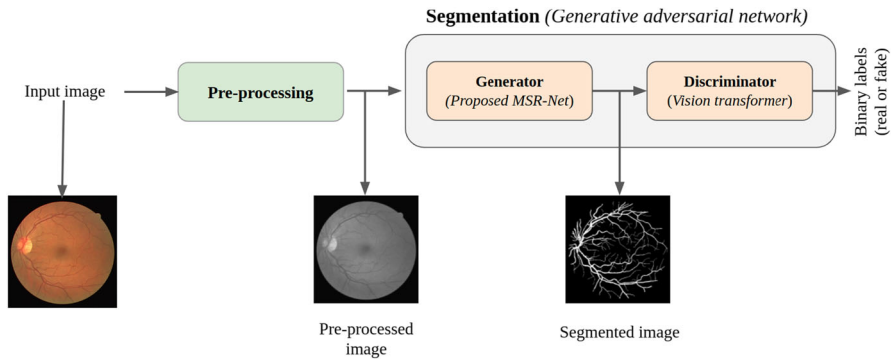


Fig. 1 Network architecture for vessel segmentation

with a generator and a discriminator. The block diagram of the proposed method is shown in Fig. 1.

Generator is a multi-scale residual convolutional neural network (MSR-Net) with skip connection framed in encoder–decoder framework used for generating segmentation maps. In generator, inception modules with UNet backbone is adopted with a joint loss to accomplish end-to-end segmentation and to capture fine vessel segments.

The discriminator is a binary classifier to classify real images (ground truth images of vessels) from fake images (generated vessel images from generator). The input to the discriminator is the human labeled ground truth images. Initially, the discriminator is trained with the real labels. The output of the generator is given to the discriminator and again the discriminator is trained to the fake labels. This adversarial training continues until the generator fools the discriminator in the sense both are efficient to produce best outputs. In the proposed method, vision transformer (ViT) [10] is used as discriminator.

2.1 Preprocessing

Preprocessing is performed prior to the image segmentation. Here, gray-level transformation is performed based on the visualization of blood vessels in the corresponding channels of color fundus image. Blood vessels are more predominant in green channel compared to red and blue channels [24]. Due to this, the color image is converted to gray scale. Data augmentation techniques like horizontal and vertical flipping are incorporated in preprocessing stage to make the data size appropriate for training.

To enhance the blood vessels and to reduce noise, CLAHE algorithm [32] is applied followed by Z-score normalization. Generally, adaptive histogram equalization (AHE) [40] is used to improve contrast in images by applying histogram equalization locally and then redistributing the brightness along the image. It has the capability of improving the local contrast of the image, but it excessively amplifies the small amount of noises in homogeneous regions of the image. To overcome this CLAHE is used where the contrast is limited within a fixed range by clipping the histogram at a predefined value before computing its cumulative distribution function (CDF). The CLAHE

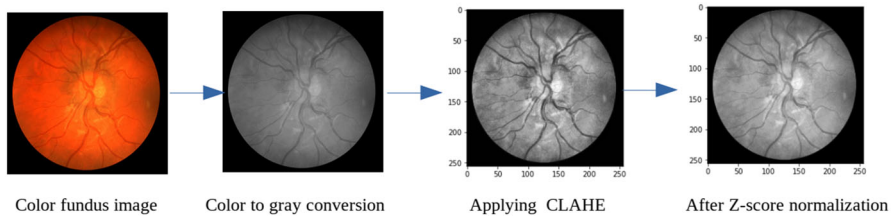


Fig. 2 Preprocessing block for enhancing the blood vessels and to reduce noise

algorithm consists of three stages: tile generation, histogram equalization, and bilinear interpolation. Initially the fundus image is partitioned into equally sized 64 rectangular sections called tiles. Histogram equalization is then performed on each tile using a contrast factor which is empirically chosen as 20, to prevent over saturation of the image. The final image is generated by combining the processed tiles by using bilinear interpolation. This effectively reduces the noises in the homogeneous non-vessel regions while enhancing the vessel pixels.

After applying CLAHE, z-score normalization is applied to reduce the noise (extra contrast) in the fundus images. The z-score normalization is given by

$$Z_i = \frac{x_i - x_{mean}}{S} \quad (1)$$

where x_i represents the i th image pixel, x_{mean} represents the mean around the pixel and S represents the standard deviation of the local patches. The preprocessing steps for enhancing blood vessels using CLAHE are shown in Fig. 2.

2.2 Network Architecture of Generator

The generator in the proposed method comprises deep convolutional network architecture with encoder–decoder structure for segmenting blood vessels from fundus image. The architecture of generator is shown in Fig. 3. Here, multi-scale convolution in UNet backbone is adopted with a joint loss to accomplish this segmentation task. The model consists of encoder stage and decoder stage. Thirty-two layers are present in the encoding section, and 25 layers are present in decoder section. The first two layers of encoder path consist of convolution layers followed by element-wise non linear activation function (ReLU) layer along with batch normalization and a max pooling layer. A bottleneck or identity module is created with inception block having residual connection. It consist of inception module with two layers of convolution followed by ReLU and batch normalization. The output of the residual module is concatenated with its input to preserve the activation from previous layer. After the each residual block again a set of convolution layer followed by ReLU, batch normalization and dropout are applied to preserve the fine details corresponding to vessel segments or pixels. This residual block with inception module is repeated for five times. Dropout is applied after each inception module to reduce complexity of CNN network and to utilize the dominant features.

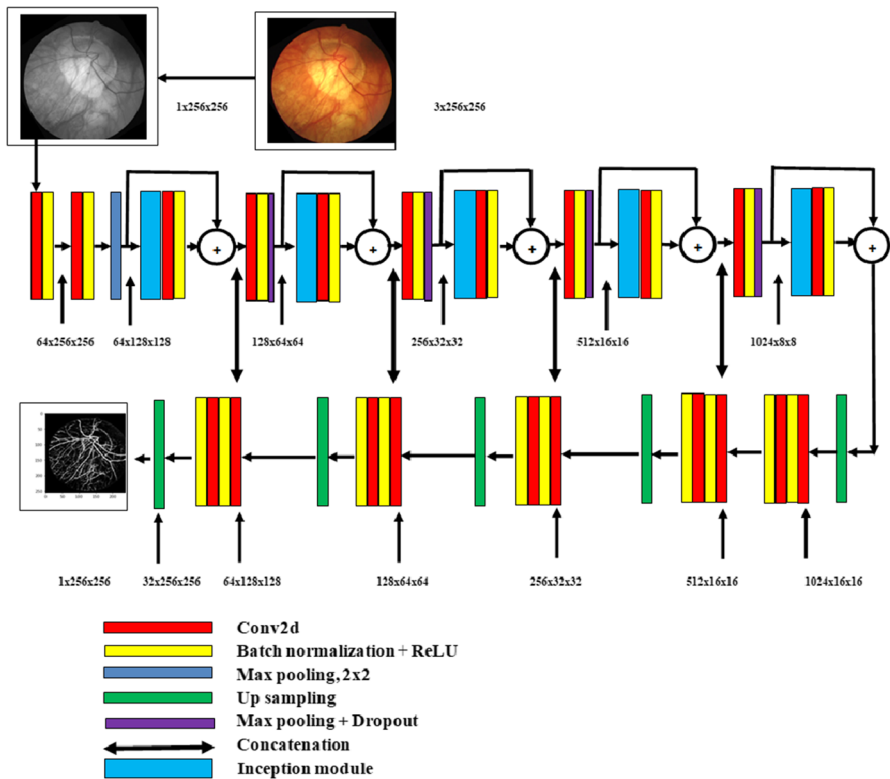


Fig. 3 Network architecture for vessel segmentation

In the decoder path, the activation’s are up-sampled with de-convolution and concatenated with the layers of encoder path of same size. This provides the positional re-occurrence of the segmented feature. Likewise, there are five units having series of convolutional layer followed by ReLU and batch normalization layer are used in the expansion path. Each unit is up-sampled with an up-sampling layer and concatenated with the encoder section. It makes use of feature maps from the lower level and with the relevant contracting path.

Batch normalization is used to reduce internal co-variate shift and provides faster convergence. It is performed by subtracting mean from the mini batch output and normalized by standard deviation of the mini batch. The batch normalized activation is given in Eq. (2).

$$A_i = \frac{a_i - a_{maen}}{\sqrt{\sigma_b^2 + c}} \tag{2}$$

where a_{maen} represents the mini batch mean, σ_b^2 is the mini batch variance, and c is a numeric constant used for numerical stability. In this case, the value of c is taken as 0.001.

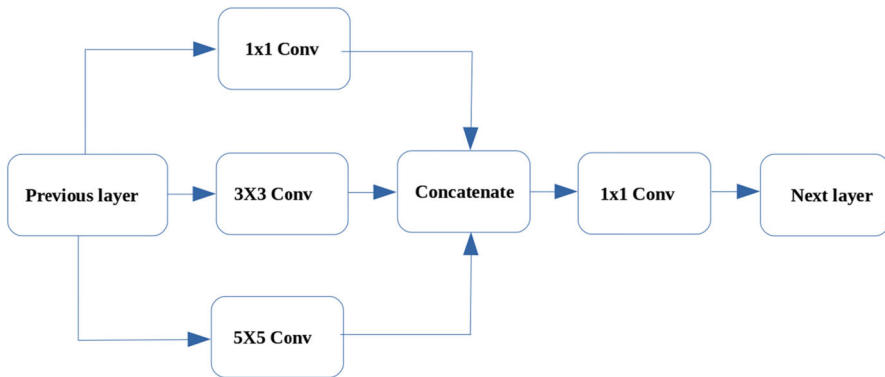


Fig. 4 Internal structure of inception module

During training the scaling and shifting parameters are updated with every epochs for faster convergence. ReLU improves the training performance by accepting the positive values only. It accelerates the computation. This follows the max pooling operation for down-sampling the data. It avoids overfitting. A dropout rate of 0.5 is employed after max pooling to reduce the overfitting.

Figure 4 shows the internal structure of inception module. It extracts features of vessel segments in different scales. Various kernel sizes, such as 1×1 , 1×3 , 3×1 , and 3×3 , are implemented to extract multi-scale features with dimension reduction. This module expands representational bottleneck by preserving the loss of information due to deeper network.

The vessel and non-vessel pixels of the fundus image are not evenly distributed. This imbalanced distribution can be overcome by using weighted binary cross-entropy (WBCE) loss function along with dice loss. The WBCE loss is defined in Eq. (3):

$$L_{WBCE}(y, \hat{y}) = -(y\beta \log(\hat{y}) + (1 - y) \log(1 - (\hat{y}))) \quad (3)$$

Here, y is the actual input to the prediction model, \hat{y} represents the predicted value by the prediction model, and β is used to tune false negatives and false positives.

The dice loss function measures similarity between true value and predicted value. It is represented in Eq. (4):

$$L_{Dice}(y, \hat{y}) = 1 - \frac{2y\hat{y} + 1}{y + \hat{y} + 1} \quad (4)$$

Here, the extra focal loss is added to the generator for minimizing the imbalance between foreground and background classes during training. The focal loss is defined as

$$L_{FOCAL}(p_t) = -(1 - p_t)^\gamma \log(p_t) \quad (5)$$

where,

$$p_t = \begin{cases} p, & \text{if } y = 1 \\ 1 - p & \text{otherwise} \end{cases}$$

$p \in [0, 1]$ is the model’s estimated probability for the class with label $y = 1$. The joint segmentation loss function for the generator is represented in Eq. (6).

$$L_{seg} = L_{WBCE} + L_{Dice}(y, \hat{y}) + L_{FOCAL} \tag{6}$$

2.3 Network Architecture of Discriminator as Transformer

In the proposed generative adversarial model, vision transformer (ViT) [10] is utilized as a discriminator. It act as a binary classifier which consists of stacked self-attention networks [44] and position-wise fully connected feed-forward networks. The transformer encoder consists of alternating layers of multi-headed self-attention and multilayer perceptron (MLP) blocks initialized with layer normalization [39]. A stack of six identical layers ($N = 6$) are used where each layer has two sublayers that is a multi-head self-attention mechanism and position-wise fully connected feed-forward networks. Residual connection [19] is utilized around each of the two sublayers, followed by layer normalization (LN) [3]. That is, the output of each sublayer is added with the input of sublayer followed by layer normalization. It can be represented by $\text{LayerNorm}(X + \text{Sublayer}(X))$, where X is the input to the sublayer. The output of this projection is mentioned as the patch embeddings. The architecture of the discriminator model is shown in Fig. 5.

The input image of size $(H \times W)$ is sampled to L patches of size $a \times a$ such that $L = (H \times W)/a^2$, which then flattened and converted to input sequences of dimension D with a trainable linear projection for the transformer encoder. These sequences can

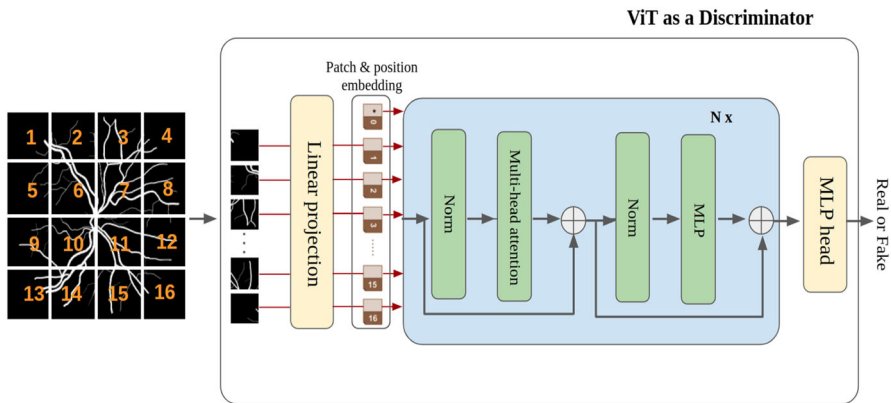


Fig. 5 Discriminator architecture for GAN

be represented by vectors as follows:

$$\text{Sublayer}_0 : s_0 = [s_0^0; \eta^1 E, \dots, \eta^L E] + E_{pos} \quad (7)$$

where linear projection $E \in \mathbb{R}^{(a^2 \cdot C) \times D}$, position embedding $E_{pos} \in \mathbb{R}^{(L+1) \times D}$, η^i is the i th patch.

$$\text{Sublayer}_i : s'_i = \text{MSA}(\text{LN}(s_{i-1})) + s_{i-1} \quad (8)$$

where $i = 1, \dots, L$ and MSA is multi-headed self-attention layer [39] and LN is the layer normalization.

$$\text{Sublayer}_{i+1} : s_i = \text{MLP}(\text{LN}(s'_i)) + s'_i \quad (9)$$

where $i = 1, \dots, L$, MLP is multilayer perceptron layer.

$$\text{Output} : y = \text{LN}(s_L^0). \quad (10)$$

The self-attention network converts the input embedding patches to three vectors called query vector (Q), key vector (K), and value vector (V). These vectors are updated during training process. The value vector is computed with positional information and the weighted sum of the value vectors gives the output of the encoder. The output of a self-attention layer is given by

$$S_0 = \text{Softmax} \left(\frac{Q \times K^T}{\sqrt{d_k}} \right) \times V \quad (11)$$

Multi-head attention allows the model to jointly attend the information from different representation sub-spaces at different positions. It has keys, values, and queries with d_k , d_v , and d_q dimensions, respectively. They are linearly projected and operated in parallel to perform the attention function, yielding a d_v dimensional output. The outputs of each head are concatenated and projected that results in the final output.

The work presented in this paper employs $h = 16$ parallel heads. For each of these, we use $d_k = d_v = 1024$. The reduction in the dimension makes the computation comparable to single-head attention with full dimensionality. The MLP contains two layers with a ReLU nonlinearity.

2.4 Network Architecture of GAN

GAN estimates generative models through an adversarial process, and it trains two models simultaneously: a generative model G for estimating the data distribution and a discriminative model D for capturing the probability that data came from the training data rather than G . The G and D play the following min-max two-player game with

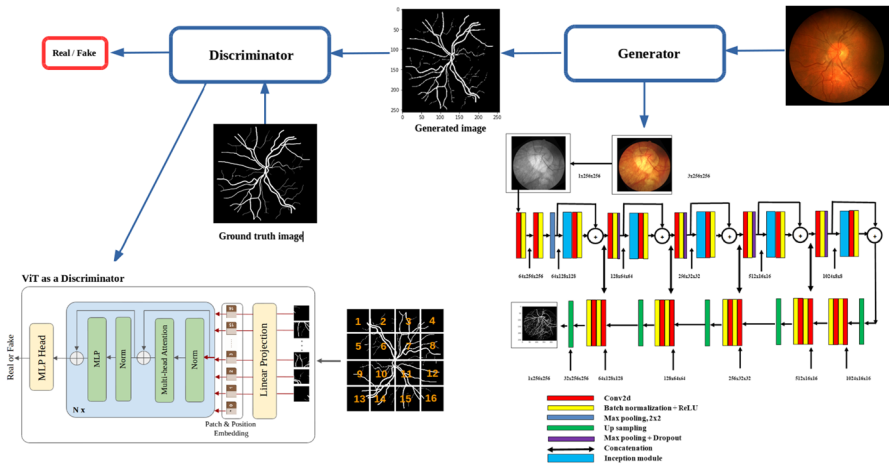


Fig. 6 Network architecture for GAN

value function $V(D, G)$ [16]. The objective function of GAN is defined as:

$$\min_{\{G\}} \max_{\{D\}} V(D, G) = E_{x \sim p_{data}(x)} [\log D(x)] + E_{z \sim p_z(z)} [\log(1 - D(G(z)))] \tag{12}$$

where x is the input to the discriminator and $D(x)$ is the output of the discriminator which is a scalar quantity. $p_{data}(x)$ is the generator’s distribution over the data x . $p_z(z)$ is the prior input noise variable. $D(x)$ denotes the probability that y was from the data rather than the generator G . The discriminator D is trained to maximize the probability of giving the correct label to both training data and fake samples generated from the generator G . G is trained to minimize $\log(1 - D(G(z)))$ simultaneously.

In this proposed method, deep convolutional GAN is utilized where both the generator and discriminator are deep neural networks. The network architecture of the proposed GAN model is shown in Fig. 6.

2.5 Loss Function of the Proposed GAN Network

The loss function of GAN consists of generator loss, discriminator loss, and GAN loss. The GAN loss consists of generator loss and discriminator loss. The generator loss is given by

$$L_{GEN} = L_{WBCE} + L_{Dice} + L_{FOCAL} = L_{GAN}(G) \tag{13}$$

The discriminator loss is given by

$$L_{DIS} = L_{BCE} = L_{GAN}(D) \tag{14}$$

In the training stage, the loss function of GAN consists of two separate functions which are given as

$$\min_{\{G\}} L_{GAN}(G) = \frac{1}{2} E_x[(1 - D(G(x)))^2] \quad (15)$$

$$\min_{\{D\}} L_{GAN}(D) = \frac{1}{2} E_y[(1 - D(y))^2] + \frac{1}{2} E_x[D(G(x))^2] \quad (16)$$

where x is the real fundus image and y is the ground truth mask. The generator of the GAN utilizes three loss functions to produce the segmented blood vessel mask. The WBCE loss function provides pixel-wise comparison of real image and ground truth where the weighting factor β is used to reduce false negatives by maximizing the true class probability.

The alternate training method is adopted to train the generative adversarial network: **Step 1:** Initially, the discriminator, which acts as a binary classifier, is trained with BCE loss function. The ground truth masks corresponding to the real samples in the dataset and the output of the generator (with frozen weights initialized randomly) are given as inputs to the discriminator and it is trained to classify them as real and fake, respectively.

Step 2: Now, the generator is trained with fundus images as inputs and corresponding masks with segmented vessels as outputs. A joint loss function is used for training. The output of the generator is fed to the discriminator. The weights of the discriminator are frozen with the weights estimated from the previous step during the training of the generator.

Step 3: *Step 1* is now repeated but the generator weights are frozen with weights estimated in *Step 2*. If the discriminator classifies both the real and fake samples with the same probability, we terminate the training, otherwise it will continue.

The pseudocode of alternate training is shown in Fig. 7.

3 Experimental Results and Analysis

All experiments in this research work are accomplished with the following: Linux operating system, Intel (R) Core (TM) CPU @ 1.8 GHz, 16 GB RAM, and GTX 1660 GPU card. Then, the system is tested using publicly available database and the performance of the proposed model has been evaluated by using different performance measures. The details about databases, experimental results, performance measures, and analysis are conferred in the subsequent subsections.

3.1 Databases

A number of publicly available retinal database have been used by the researchers for segmentation of blood vessels and detection of retinal diseases. Mainly researches have used Digital Retinal Images for Vessel Extraction (DRIVE) database [29], Structured Analysis of the Retina (STARE) database [21], and CHASE-DB1 [30] database for

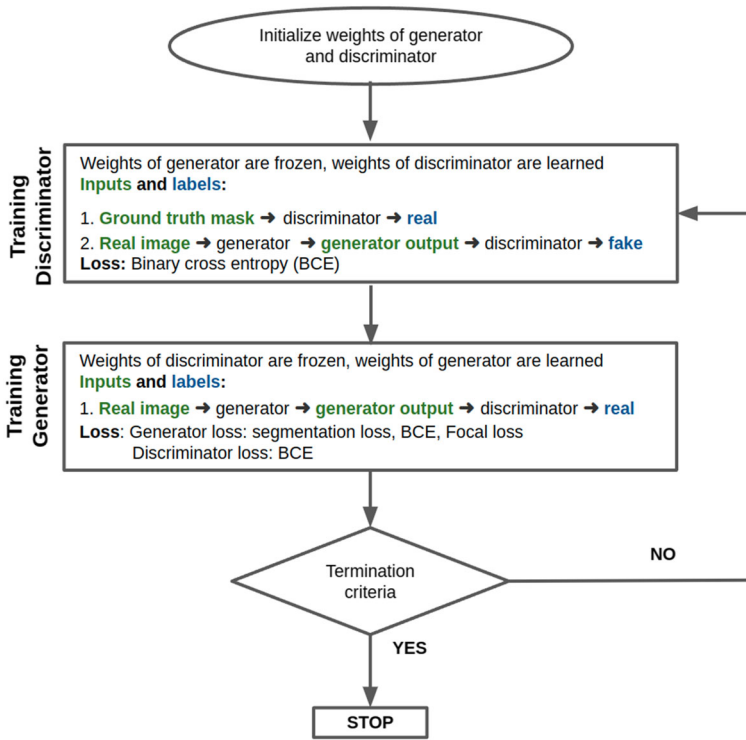


Fig. 7 Training algorithm for GAN

segmentation of blood vessels. In this paper, seven databases have been used for analyzing the performance of the proposed method.

DRIVE database comprises 40 images as vasculature ground truth, of which 20 images are employed as training set and rest as the testing set. All the images are captured with a 45 degree field of view (FOV), having resolution of 565×584 pixels. With the test database, two manually segmented ground truths are provided, where one set is used as gold standard and the other can be used to evaluate computer generated segmentation for comparison.

STARE database contains 20 images with two sets of manually labeled vessel segmented ground truths. These are captured with 35 degree field of view and having 700×605 pixel resolution. Among the two sets, the first one is labeled as ground truth whereas the other is taken as gold standard.

The CHASE_DB1 database comprises of 14 pairs of retinal fundus images having a resolution 960×999 pixels with a 30 deg FOV, collected from multiethnic school children [30] in London. In the database, ground truth vessel annotations are available in two sets. The first set is normally used for training and testing, whereas the second one acts as a human baseline.

The HRF database [5] contains 15 images of healthy patients, 15 images of patients with diabetic retinopathy and 15 images of glaucomatous patients. Binary gold stan-

standard vessel segmentation images are available for each image. The gold standard data is generated by a group of experts working in the field of retinal image analysis and clinicians from the cooperated ophthalmology clinics.

The Automated Retinal Image Analysis (ARIA) database [13] for vessel extraction, the ARIA database consists of 138 images taken either from healthy subjects, diabetics, or from patients with age-related macular degeneration (AMD). All of these images were collected with a Zeiss FF450+ fundus camera with a 50 degree angular field of view (FOV).

The RC-SLO dataset [8] contains 40 image patches with a resolution of 360×320 pixels that are annotated by experts. This dataset covers a wide range of difficult cases, such as high curvature changes, central vessel reflex, micro-vessels, crossings/bifurcations and background artifacts. The images in the RC-SLO dataset are acquired with an EasyScan camera (i-Optics Inc., the Netherlands), which is based on a scanning laser ophthalmoscopy (SLO) technique.

The IOSTAR vessel segmentation dataset [45] consists of 30 images with a resolution of 1024×1024 pixels. This dataset is the developed version of Scanning Laser Ophthalmoscopy (SLO) images. The images in the IOSTAR vessel segmentation dataset are captured with an EasyScan camera, which is based on a SLO technique with a 45 degree FOV. All the ground truth images of vessels in this dataset are annotated by a group of experts working in the field of retinal image analysis.

3.2 Performance Measures

Various performance measures have been suggested by the researchers to evaluate the segmentation accuracy of the deep learning models. These performance measures are based on the number of correctly segmented vessels and non-vessel pixels.

The correctly segmented vessel pixels are treated as true positives (TP_v), whereas wrongly segmented vessel pixels are called false negatives (FN_v). True negative (TN_{nv}) and false positive (FP_{nv}) represent the correctly segmented non-vessel pixels and incorrectly segmented non-vessel pixels, respectively. These TP_v , FP_{nv} , FN_v , and TN_{nv} are used to calculate the different performance measures such as accuracy (Acc), sensitivity (Sen), specificity (Spe), precision (Pre), $F1Score$, intersection over union (IOU), and area under the ROC curve (AUC) etc [11].

Accuracy (Acc) can be defined as the ratio of correctly segmented blood vessel pixels to the total number of pixels in the image. It is the gold standard metric for all types segmentation. Specificity (Spe) measures the correct segmented non-vessel pixels. Sensitivity (Sen) is a measure to compute the segmented vessel pixels correctly by the model. Precision (Pre) measures the fraction of perfectly segmented blood vessel pixels to the total number of segmented blood vessel pixels. These measures are represented in Eqs. (17) to 20:

$$Acc = \frac{TP_v + TN_{nv}}{TP_v + TN_{nv} + FP_{nv} + FN_v} \times 100 \quad (17)$$

$$Sen = \frac{TP_v}{TP_v + FN_v} \times 100 \quad (18)$$

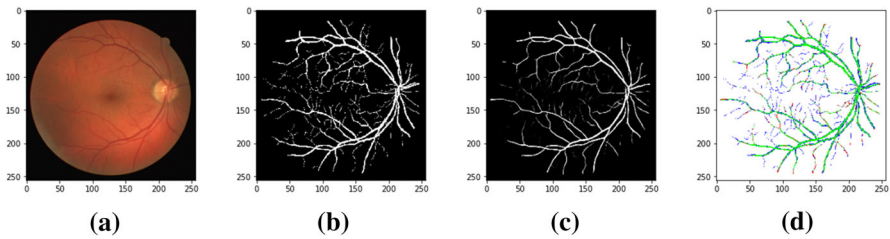


Fig. 8 Segmentation result of the proposed algorithm for DRIVE database: **a** color fundus image, **b** ground truth image, **c** generator output image, and **d** visual difference image (Color figure online)

$$Spe = \frac{TN_{nv}}{TN_{nv} + FP_{nv}} \times 100 \quad (19)$$

$$Pre = \frac{TP_v}{TP_v + FP_{nv}} \times 100 \quad (20)$$

On the basis of the obtained segmented heat map, the receiver operating characteristic (ROC) curve is found out. The area under the ROC curve (*AUC*) is used as quantitative indicators for analyzing the segmentation. In addition, *F1Score* is often presented in papers as one of the evaluation criteria for segmentation. *F1Score* is defined as

$$F1Score = \frac{2TP_v}{2TP_v + FN_v + FP_{nv}} \times 100 \quad (21)$$

The *IOU* is defined as:

$$IOU = \frac{TP_v}{TP_v + FN_v + FP_{nv}} \quad (22)$$

3.3 Results

The model is trained by Adam optimizer with learning rate 0.0002 and batch size 8. The total training process complied with 150 epochs. The limit of the best epoch can be approximately determined by observing the training accuracy curve in training dataset and loss in validation dataset during training. The model with the best performance is chosen as the final trained model.

The publicly available databases discussed in Sect. 3.1 have been used by the network, and various performance measures (such as *Acc*, *Sen*, *Spe*, *Pre*, *F1Score*, and *IOU*) are computed and analyzed. Before feeding the databases into the network they are preprocessed (resized and augmented). During the augmentation process (vertical flipping, horizontal flipping, and rotation), the number of images present in the databases is increased. The model is trained according to the proposed training algorithm shown in Fig. 7. The segmentation results of the algorithm proposed in this paper are shown in Figs. 8, 9, 10, 11, 12, 13, and 14. Figure 8 shows the segmentation

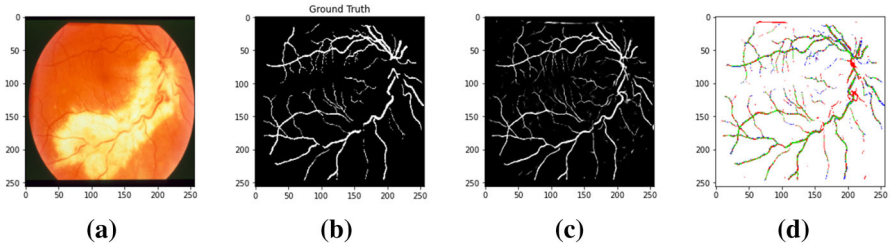


Fig. 9 Segmentation result of the proposed algorithm for STARE database: **a** color fundus image, **b** ground truth image, **c** generator output image, and **d** visual difference image (Color figure online)

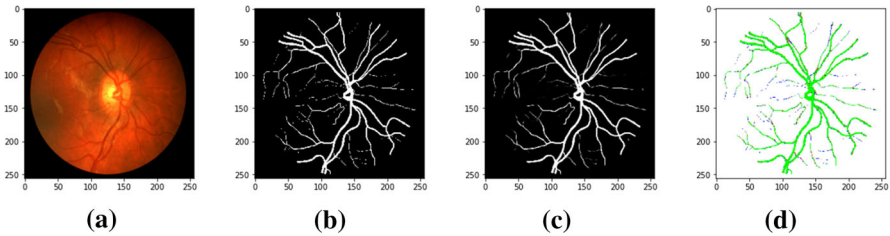


Fig. 10 Segmentation result of the proposed algorithm for CHASE_DB1 database: **a** color fundus image, **b** ground truth image, **c** generator output image, and **d** visual difference image (Color figure online)

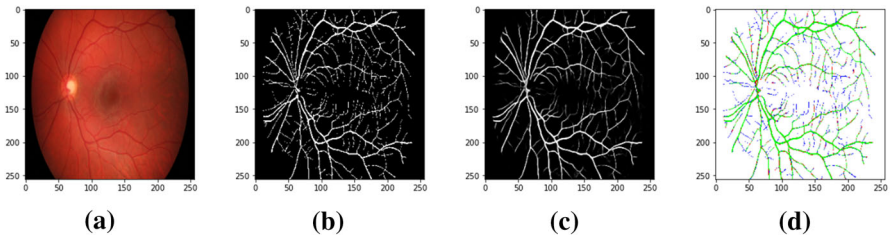


Fig. 11 Segmentation result of the proposed algorithm for HRF database: **a** color fundus image, **b** ground truth image, **c** generator output image, and **d** visual difference image (Color figure online)

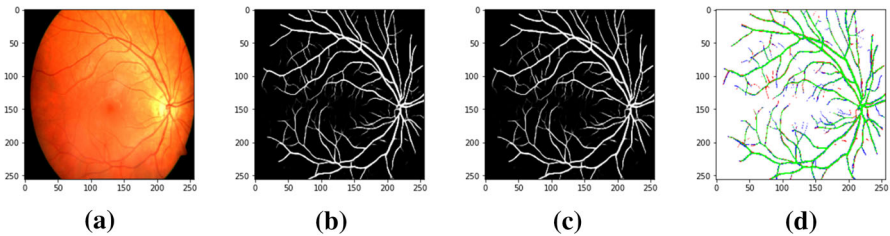


Fig. 12 Segmentation result of the proposed algorithm for ARIA database: **a** color fundus image, **b** ground truth image, **c** generator output image, and **d** visual difference image (Color figure online)

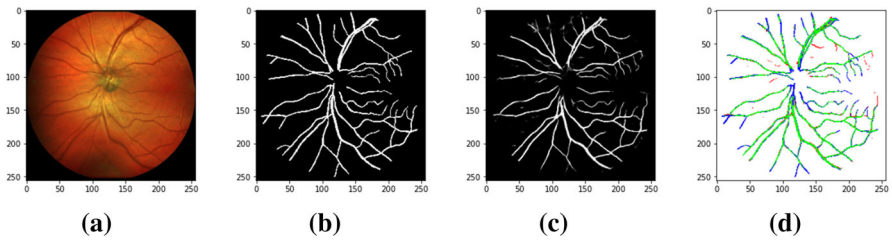


Fig. 13 Segmentation result of the proposed algorithm for IOSTAR database: **a** color fundus image, **b** ground truth image, **c** generator output image, and **d** visual difference image (Color figure online)

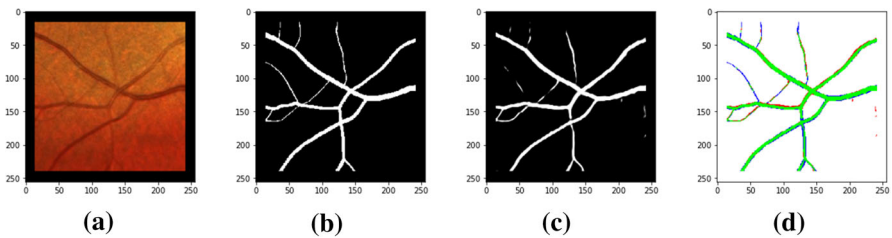


Fig. 14 Segmentation result of the proposed algorithm for RC-SLO database: **a** color fundus image, **b** ground truth image, **c** generator output image, and **d** visual difference image (Color figure online)

result of an image from DRIVE database. Figure 8a represents the resized original color fundus image, Fig. 8b represents the ground truth image, Fig. 8c represents generator output image, and Fig. 8d represents the visual difference image, which shows the visual difference between the generated and ground truth images in terms of different colors. In the visual difference image, green, white, red, and blue represent TP, TN, FP, and FN, respectively. Similarly Figs. 9, 10, 11, 12, 13, and 14 represent the segmentation results from STARE, CHASE_DB1, HRF, ARIA, IOSTAR, and RC-SLO databases, respectively.

The proposed method achieved the highest accuracy of 0.9742 for DRIVE database, 0.9486 for STARE database, and 0.9873 for CHASE_DB1 database compared to the recorded values in the literature. Also, the proposed method shows the highest *AUC* score for CHASE_DB1 database (0.9880). The proposed method achieved the highest *IOU* for DRIVE and CHASE_DB1 databases. The overall performance measures for mentioned databases are shown in Table 1.

The visual results of the proposed method are compared with different segmentation methods, which are shown in Figs. 15 and 16. The segmentation result of the proposed method is compared with UNet model [33], UNet++ model [48], and deep vessel model [15]. The proposed GAN model with transformer as discriminator is more effective in segmenting blood vessels from fundus images by using multiple loss functions. In Fig. 15, the first row represents the highlighted parts of the generated output of different models along with proposed model and the ground truth. The second row represents the magnified parts of the generated visual difference images of different models along with proposed model. The third row represents the visual difference images of different models along with the ground truth image. In Fig. 16, the first column represents the ground truth image with highlighted parts and visual difference

Table 1 Recorded performance measures for the segmented blood vessels for different databases in the literature

Dataset	Method	<i>Acc</i>	<i>Spe</i>	<i>Sen</i>	<i>Pre</i>	<i>AUC</i>	<i>IOU</i>	
DRIVE	CNN (Uysal et al. 2020) [38]	0.9527	0.9784	0.7778	0.8417	–	–	
	DeepVessel (Fu et al. 2016) [15]	0.9523	–	0.7603	–	–	–	
	CNN+GNN (Shin et al. 2019) [36]	0.9271	0.9255	0.9382	–	0.9802	–	
	SegmentLoss (Yan et al. 2018) [42]	0.9542	0.9818	0.7653	–	0.9752	–	
	DUNet (Jin et al. 2019) [23]	0.9697	–	–	0.8537	–	–	
	CE-Net (Gu et al. 2019) [17]	0.9545	–	0.8309	–	0.9779	–	
	GAN + Dense UNet (Guo et al. 2020) [18]	0.9542	0.9726	0.8283	–	0.9772	0.8251	
	M-GAN (Park et al. 2020) [31]	0.9706	0.9836	0.8346	0.8302	0.9868	0.7129	
	Bridge-Net (Zhang et al. 2019) [46]	0.9565	0.9818	0.7853	0.8635	0.9834	–	
	D-MNet (Xiangyu et al. 2022) [9]	0.9683	0.9811	0.8363	–	–	–	
PCAT-UNet (Danny et al. 2022) [6]	0.9622	0.9932	0.8576	–	0.9872	–		
	Proposed method	0.9742	0.9876	0.8936	0.8748	0.9650	0.8580	
STARE	CNN (Uysal et al. 2020) [38]	0.9589	0.9811	0.7558	0.8231	–	–	
	DeepVessel (Fu et al. 2016) [15]	0.9585	–	0.7412	–	–	–	
	CNN+GNN (Shin et al. 2019) [36]	0.9378	0.9352	0.9598	–	0.9877	–	
	SegmentLoss (Yan et al. 2018) [42]	0.9612	0.9846	0.7581	–	0.9801	–	
	CE-Net (Gu et al. 2019) [17]	0.8490	0.7650	0.8530	0.8050	0.8530	0.8530	
	DUNet (Jin et al. 2019) [23]	0.9729	–	–	0.8856	–	–	
	M-GAN (Park et al. 2020) [31]	0.9876	0.9938	0.8324	0.8417	0.9873	0.7189	
	Bridge-Net (Zhang et al. 2019) [46]	0.9668	0.9864	0.8002	0.8711	0.9901	–	
	D-MNet (Xiangyu et al. 2022) [9]	0.9643	0.9847	0.8272	–	–	–	
	PCAT-UNet (Danny et al. 2022) [6]	0.9796	0.9937	0.8703	–	0.9953	–	
	Proposed method	0.9486	0.9640	0.8441	0.8212	0.9532	0.7580	
CHASE_DB1	DeepVessel (Fu et al. 2016) [15]	0.9489	–	0.7130	–	–	–	
	CNN+GNN (Shin et al. 2019) [36]	0.9373	0.9364	0.9463	–	0.9830	–	
	SegmentLoss (Yan et al. 2018) [42]	0.9610	0.9809	0.7633	–	0.9781	–	
	DUNet (Jin et al., 2019) [23]	0.9724	–	–	0.7510	0.9863	–	
	M-GAN (Park et al. 2020) [31]	0.9736	–	–	–	0.9859	–	
	Bridge-Net (Zhang et al. 2019) [46]	0.9667	0.9840	0.8132	0.8490	0.9893	–	
	D-MNet (Xiangyu et al. 2022) [9]	0.9714	0.9794	0.8541	–	–	–	
	PCAT-UNet (Danny et al. 2022) [6]	0.9812	0.9966	0.8493	–	0.9925	–	
		Proposed method	0.9873	0.9835	0.9335	0.9154	0.9880	0.8792
		CNN+GNN (Shin et al. 2019) [36]	0.9349	0.9329	0.9546	–	0.9838	–
HRF	SegmentLoss (Yan et al. 2018) [42]	0.9437	0.9809	0.7633	–	0.9781	–	
	Bridge-Net (Zhang et al. 2019) [46]	0.9590	0.9690	0.8570	–	–	–	
	M-GAN (Park et al. 2020) [31]	0.9761	–	–	–	0.9859	–	
	D-MNet (Xiangyu et al. 2022) [9]	0.9668	0.9848	0.7543	–	–	–	
		Proposed method	0.9773	0.9854	0.8885	0.8003	0.9920	0.7037

Table 1 continued

Dataset	Method	<i>Acc</i>	<i>Spe</i>	<i>Sen</i>	<i>Pre</i>	<i>AUC</i>	<i>IOU</i>
ARIA	Graph Cut Approach (Zhao et al. 2015) [47]	0.9400	0.9300	0.7510	–	0.8410	–
	Cognitive Graph-Based (Shehhi et al. 2016) [35]	0.8190	0.8230	0.7810	–	0.8020	–
	Proposed method	0.9628	0.9840	0.7177	0.7950	0.9671	0.7492
IOSTAR	Adaptive derivatives (Zhang et al. 2016) [45]	0.9514	0.9740	0.7545	–	0.9615	–
	Proposed method	0.9610	0.9788	0.7807	0.7847	0.9767	0.6492
RC-SLO	Adaptive derivatives (Zhang et al. 2016) [45]	0.9512	0.9710	0.7787	–	0.9626	–
	Proposed method	0.9777	0.9871	0.8714	0.8381	0.9940	0.7459

image. The second column represents comparison of highlighted segmented parts of the generated output of proposed model along with visual difference image. The third column represents comparison of highlighted segmented parts of the generated output of UNet++ model along with visual difference image. The fourth column represents comparison of highlighted segmented parts of the generated output of DeepVessel model along with visual difference image and the fifth column represents comparison of highlighted segmented parts of the generated output of UNet model along with visual difference image.

Table 1 shows a relative performance measure of recorded values in the literature works for retinal vessel segmentation, evaluated on publicly available datasets. Mainly, UNet-based methods and its extended interpretations are implemented for retinal blood vessel segmentation purposes using different loss functions [38–42]. Authors mainly focused on calculating accuracy, precision, sensitivity, and specificity, which demonstrate the quality of segmentation. Uysal et al. [38] used fully connected CNN for vessel segmentation where they achieved an accuracy of 95.27% and sensitivity of 77.78% in case of DRIVE database. Similarly, Fu et al. [15] implemented multi-scale and multi-level CNN with conditional random field to model the network (DeepVessel) for vessel segmentation. They got the highest accuracy of 95.85% for STARE database. Shin et al. [36] implemented graph neural network (GNN) together with CNN to segment the blood vessels from fundus images. They have used four retinal fundus image databases namely DRIVE [29], STARE [21], CHASE_DB1 [30], and HRF [14] databases to testify their proposed method. They achieved the highest *AUC* of 98.38% for HRF database. Similarly, Yan et al. [42] achieved the highest specificity of 98.46% for STARE dataset by incorporating segment-level and the pixel-wise losses into deep CNN model. But recently introduction of GAN into vessel segmentation task gives better performance compared to UNet-based segmentation models. Guo et al. [18] proposed the GAN architecture where the generator is a dense UNet using the inception module and the discriminator is a deep neural network used as a binary classifier. They achieved the highest *AUC* value of 0.9772 and *F1Score* of

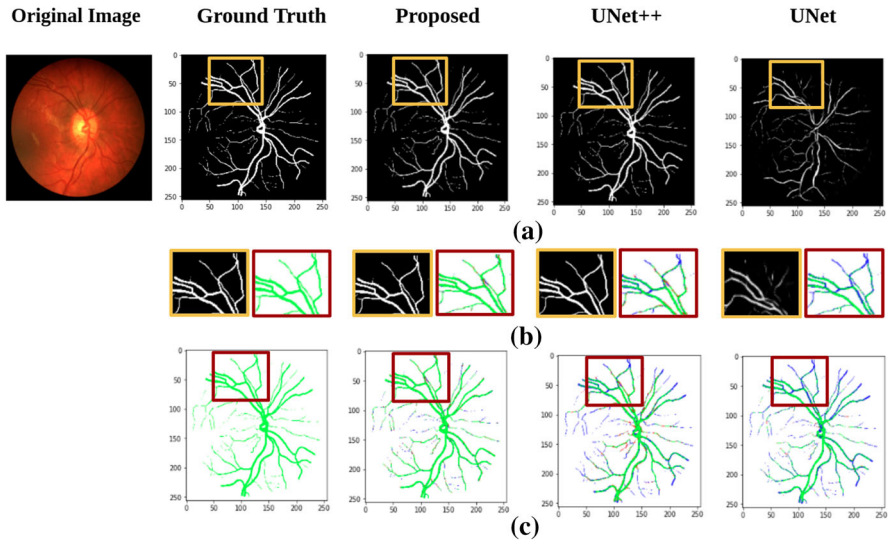


Fig. 15 Comparison of visual results for CHASE_DB1 dataset: **a** comparison of segmented output by the proposed method with ground truth and other models, **b** comparison of magnified segmented blood vessels by proposed method and other methods, and **c** comparison of visual difference images of segmented output of the proposed method and other methods

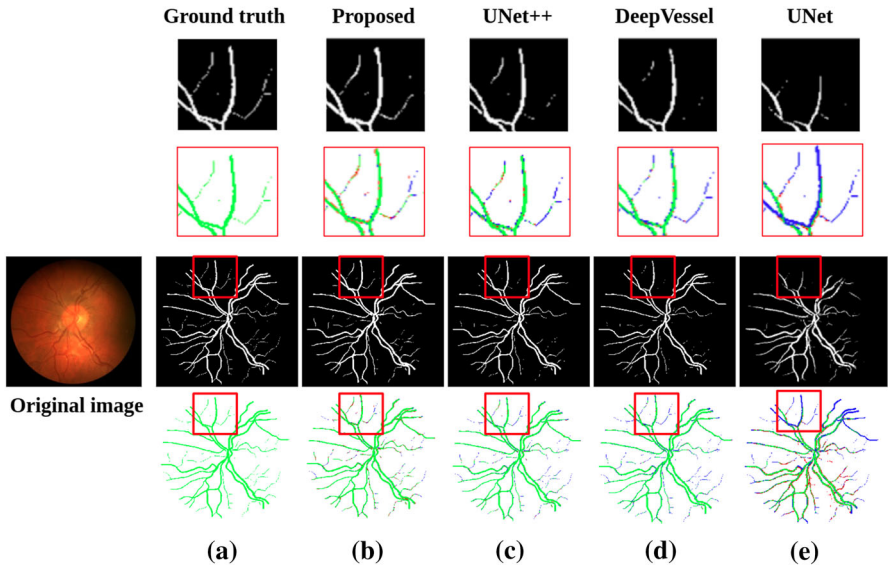


Fig. 16 Comparison of visual results for CHASE_DB1 dataset: **a** ground truth image with highlighted parts and visual difference image, **b** comparison of highlighted segmented parts of the generated output of proposed model along with visual difference image, **c** comparison of highlighted segmented parts of the generated output of UNet++ model along with visual difference image, **d** comparison of highlighted segmented parts of the generated output of DeepVessel model along with visual difference image, **e** comparison of highlighted segmented parts of the generated output of UNet model along with visual difference image

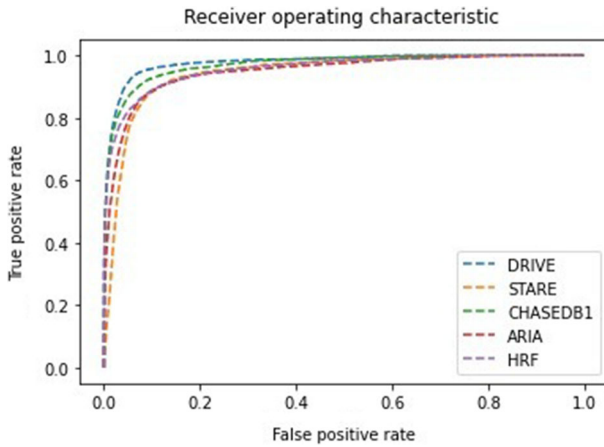


Fig. 17 Receiver operating characteristics (ROC) for DRIVE, STARE, CHASE_DB1, ARIA, and HRF database

0.8215 on DRIVE database. Similarly, Park et al. [31] proposed M-GAN model for vessel segmentation which achieved an average accuracy of 97.06%, AUC of 98.68%, and *F1Score* of 0.8317.

The proposed approach exceeds previous approaches with respect to the accuracy, *IoU*, and AUC. We have compared the proposed MSR-GAN with recorded values of related previous studies. The comparative evaluation using the DRIVE dataset, STARE database, CHASE_DB1 database, HRF database, ARIA database, IOSTAR database, and RC-SLO database are described in Table 1. MSR-GAN model showed higher performance than related studies concerning accuracy, *IoU*, and AUC measurements.

The ROC curve for the above-mentioned databases is shown in Fig. 17. It is observed that the proposed method achieved the highest AUC for CHASE_DB1 database (green dashed line) and lowest for STARE database (orange dashed line). It shows the relationship between true positive rate and false positive rate.

4 Ablation Study Using Different Loss Functions and Different Patch Sizes on the Proposed Model

To affirm the efficacy of the proposed MSR-GAN architecture, different conditions are imposed utilizing individual loss functions on the model. At first, experiments are conducted using WBCE loss function to the generator. The generator is trained with WBCE loss and GAN loss and the performance measures are calculated. Then again, the generator is trained with WBCE loss and dice loss along with GAN loss and the performance measures are given in Table 2. The performance measures are calculated using DRIVE, STARE, and CHASE_DB1 databases. It is observed from Table 2 that the proposed GAN network, the intersection over union (*IoU*), and *F1Score* increase apparently by using GAN loss.

Table 2 Performance metrics for segmented blood vessels from retinal images using different databases

Dataset	Loss Function	<i>IOU</i>	<i>F1Score</i>
DRIVE	UNet + BCE loss	0.7490	0.7950
	MSR-Net + WBCE loss + DICE loss	0.8280	0.8200
	MSR-Net + WBCE loss + DICE loss + Focal loss	0.8380	0.8560
	MSR-Net + GAN loss	0.8580	0.8734
	UNet + BCE loss	0.6490	0.7650
STARE	MSR-Net + WBCE loss + DICE loss	0.6980	0.8097
	MSR-Net + WBCE loss + DICE loss + Focal loss	0.7380	0.8163
	MSR-Net + GAN loss	0.7580	0.8200
	UNet + BCE loss	0.7576	0.7858
CHASE_DB1	MSR-Net + WBCE loss + DICE loss	0.8582	0.8480
	MSR-Net+WBCE loss + DICE loss + Focal loss	0.8580	0.8508
	MSR-Net + GAN loss	0.8786	0.8889

Table 3 Performance metrics for segmented blood vessels from retinal images in DRIVE database

Model	<i>Acc</i>	<i>Sen</i>	<i>Spe</i>	<i>IOU</i>
UNet	0.8490	0.7650	0.8530	0.7490
MSR-Net	0.7980	0.6500	0.7630	0.8380
MSR-Net+GAN	0.9873	0.8936	0.9876	0.8580

Table 3 presents the performance of different models for DRIVE database. Compared with the classic UNet, the MSR-Net performed better in *Acc*, *Sen*, *Spe*, and *F1Score*. But using GAN its performance increases furthermore which verifies the effectiveness of our proposed algorithm.

In order to justify the potentiality of the model, the experiments are conducted with different patch sizes a using the DRIVE, STARE and CHASE_DB1 databases. Initially the results are tabulated by taking different patch sizes. It is found that the medium sized patches ($a = 32$ and 64) are giving better accuracy than smaller ($a = 16$) and larger ($a = 128$) patch sizes. The comparison of segmentation performance using different patch sizes is given in Table 4.

4.1 Ablation Study of the Proposed Model with Two Different Types of Discriminator

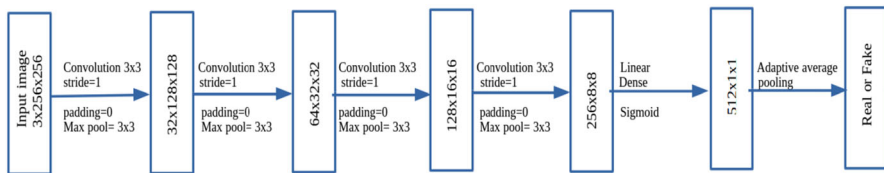
The proposed GAN architecture is experimented with two types of discriminators that is a deep CNN-based binary classifier [18] and vision transformer [39]. Both discriminators used the same binary cross-entropy loss function. The discriminator model using deep CNN architecture is a binary classifier comprised of four convolutional neural networks (CNNs) and two fully connected layers. After each convolutional layer, there is a pooling layer. The convolutional layers process the input image and extract a feature vector fed to the fully connected layers and finally the output layer is

Table 4 Comparison of segmentation results using different patch sizes

Dataset	Patch size	<i>Acc</i>	<i>Spe</i>	<i>Sen</i>	<i>Pre</i>	<i>AUC</i>	<i>IOU</i>	<i>F1Score</i>
DRIVE	16	0.9542	0.9920	0.8778	0.8800	0.9450	0.8235	0.7200
	32	0.9624	0.9915	0.8736	0.8862	0.9610	0.8280	0.7281
	64	0.9616	0.9928	0.8700	0.8853	0.9590	0.8205	0.7168
	128	0.9543	0.9888	0.8645	0.8447	0.9600	0.5654	0.7223
STARE	16	0.9600	0.9790	0.7558	0.8231	0.9450	0.8154	0.5910
	32	0.9620	0.9876	0.8936	0.7577	0.9150	0.8196	0.6058
	64	0.9709	0.9856	0.8968	0.8158	0.9300	0.8080	0.6200
CHASE_DB1	16	0.9579	0.9855	0.8626	0.8025	0.8970	0.7912	0.6733
	32	0.9812	0.9904	0.8900	0.8850	0.9930	0.8521	0.8780
	64	0.9847	0.9956	0.8909	0.9122	0.9960	0.8205	0.9014
CHASE_DB1	64	0.9750	0.9887	0.8553	0.8572	0.9900	0.7283	0.8428
	128	0.9787	0.9834	0.9291	0.8369	0.9880	0.7866	0.8806

Table 5 Comparison of segmentation results using two different types of discriminators

Dataset	Discriminator	<i>Acc</i>	<i>Spe</i>	<i>Sen</i>	<i>Pre</i>	<i>AUC</i>	<i>IOU</i>	<i>F1Score</i>
DRIVE	Deep CNN	0.9542	0.9679	0.8778	0.8248	0.9450	0.8235	0.7200
	ViT	0.9742	0.9876	0.8936	0.8748	0.9650	0.8580	0.8700
STARE	Deep CNN	0.9513	0.9511	0.7558	0.8231	0.9320	0.7554	0.5910
	ViT	0.9486	0.9640	0.8441	0.8212	0.9532	0.8154	0.6253
CHASE_DB1	Deep CNN	0.9752	0.9875	0.8900	0.8084	0.9860	0.8521	0.8708
	ViT	0.9873	0.9835	0.9335	0.9154	0.9880	0.8786	0.8889

**Fig. 18** CNN-based binary classifier as discriminator architecture of GAN (Color figure online)

used for binary classification. The model is shown in Fig. 18. It is found that vision transformer-based discriminator produces better result than CNN-based binary classifier model. Table 5 shows relative performance measures of the proposed GAN model using two types of discriminators evaluated on publicly available databases.

MSR-GAN network achieved higher accuracy and AUC in the case of the CHASE_DB1 database and higher specificity in the DRIVE database. The generator and discriminator of the GAN are alternatively trained using different loss functions to achieve the best result concerning their objectivity. The generator learns ground truth data distribution better using this adversarial training with the binary classifier as a discriminator. The proposed MSR-GAN model achieved the highest accuracy and sensitivity from CHASE_DB1 and DRIVE databases compared to other existing

GAN-based segmentation algorithms [18–31]. The network's performance is significantly improved by using combined loss functions which consist of WBCE loss, DICE loss, and GAN loss. Also, additive focal loss increases the network's performance by reducing the false negatives and false positives. The proposed architecture is also able to reduce the class imbalance that occurred between foreground and background during training. Hence, the network structure can extract deep features and perform robust retinal vessel segmentation.

The proposed approach outperformed previous research on different performance measures. The dataset performed well in the case of the CHASE_DB1 database compared to DRIVE and STARE databases. It shows the highest accuracy of 0.9873 and sensitivity of 0.9335 in the CHASE_DB1 database and DRIVE database compared to M-GAN architecture [31] and DI-UNet model with GAN [18].

5 Performance of the Proposed Method in Segmenting Low-Quality Retinal Images

In order to check the robustness of the model in the presence of noise and low-quality retinal images, the model is used on synthetic noisy images of DRIVE and RC-SLO databases. The synthetic images are generated by adding zero-mean Gaussian noise of varying standard deviation (σ_n) to the original image present in the database [25]. The generated low-quality image is presented as

$$I_\eta(i, j) = I(i, j) + \eta(i, j), \quad (23)$$

where $I(i, j)$ is the original image and $\eta(i, j)$ is zero-mean Gaussian noise whose distribution is represented by

$$P[\eta(i, j)] = \frac{1}{(2\pi)^{1/2}\sigma_n} \exp\left\{-\frac{\eta(i, j)^2}{2\sigma_n^2}\right\}. \quad (24)$$

Initially, the images are blurred by 7×7 Gaussian filter with standard deviation σ_b varying between 0.5 and 3.0. These blurred images are added with zero-mean Gaussian noise of standard deviation σ_n varying between 0.001 and 0.02. The formation of noisy images for DRIVE and RC-SLO databases is shown in Figs. 19 and 20, respectively.

Table 6 shows the performance of the proposed model for the synthetic low-quality images generated by the addition of blurred and Gaussian noise to the original data. The accuracy value is found to be lie between 95.94 and 97.93% for all types of low-quality images. These values are lies close to the segmentation accuracy value of the retinal images without degradation. From this experiment it may be concluded that the method is robust in presence of noise.

To analyze the visual appearance of the segmented blood vessels from the low-quality images (affected by noise) and the images without any noise, the original and segmented images are shown in Fig. 21. Figure 21a represents a original color image from RC-SLO database. Figure 21b–d represents the corresponding ground truth image, generator output image, and visual difference image, respectively. Simi-

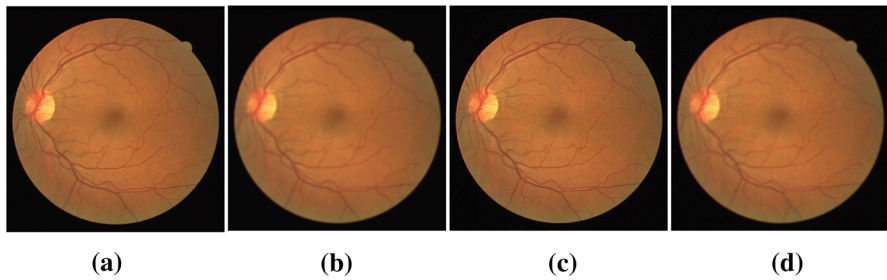


Fig. 19 Formation of noisy image from DRIVE database: **a** original color image, **b** generated blurred image ($\sigma_b = 3.0$), **c** generated noisy image ($\sigma_n = 0.02$), **d** generated blurred and noisy image (Color figure online)

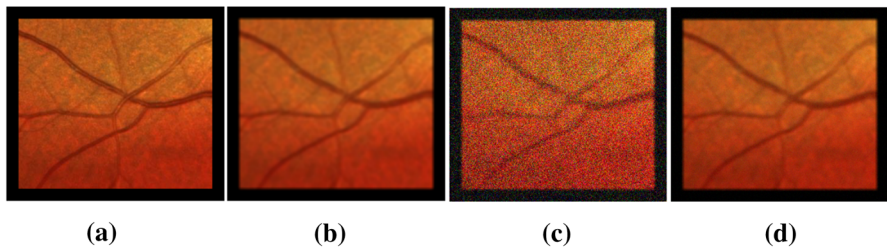


Fig. 20 Formation of noisy image from RC-SLO database: **a** original color image, **b** generated blurred image ($\sigma_b = 3.0$), **c** generated noisy image ($\sigma_n = 0.02$), **d** generated blurred and noisy image (Color figure online)

Table 6 Performance measures with added noise for DRIVE and RC-SLO databases

Dataset	σ_b	σ_n	<i>Acc</i>	<i>Spe</i>	<i>Sen</i>	<i>Pre</i>	<i>AUC</i>	<i>IOU</i>	<i>F1Score</i>
DRIVE	0.05	0.001	0.9710	0.9838	0.8936	0.8748	0.9650	0.8580	0.8720
		0.005	0.9703	0.9876	0.8936	0.8748	0.9650	0.8580	0.8694
		0.02	0.9513	0.9811	0.8858	0.8231	0.962	0.8154	0.8610
	1.5	0.001	0.9632	0.9829	0.8968	0.8512	0.9590	0.8554	0.8753
		0.005	0.9607	0.9875	0.8900	0.8584	0.964	0.8521	0.8708
		0.02	0.9594	0.9829	0.8816	0.8544	0.9400	0.8489	0.8689
	3	0.001	0.9729	0.9817	0.8797	0.8660	0.9640	0.8530	0.8561
		0.005	0.9710	0.9868	0.8736	0.8684	0.9630	0.8428	0.8538
		0.02	0.9573	0.9871	0.8635	0.8654	0.9650	0.8359	0.8509
RC-SLO	0.05	0.001	0.9787	0.9840	0.9139	0.8228	0.9950	0.7637	0.8660
		0.005	0.9793	0.9845	0.9153	0.8275	0.9950	0.7687	0.7816
		0.02	0.9750	0.9824	0.8457	0.7345	0.9870	0.6477	0.7862
	1.5	0.001	0.9616	0.9718	0.8510	0.7371	0.9780	0.6517	0.7891
		0.005	0.9769	0.9836	0.8598	0.7511	0.9780	0.6692	0.8018
		0.02	0.9766	0.9849	0.8732	0.8251	0.9840	0.7368	0.8485
	3	0.001	0.9756	0.9869	0.8066	0.7588	0.98530	0.6420	0.7819
		0.005	0.9742	0.9841	0.8521	0.8137	0.9770	0.7130	0.8324
		0.02	0.9770	0.9847	0.8822	0.8242	0.9930	0.7425	0.8522

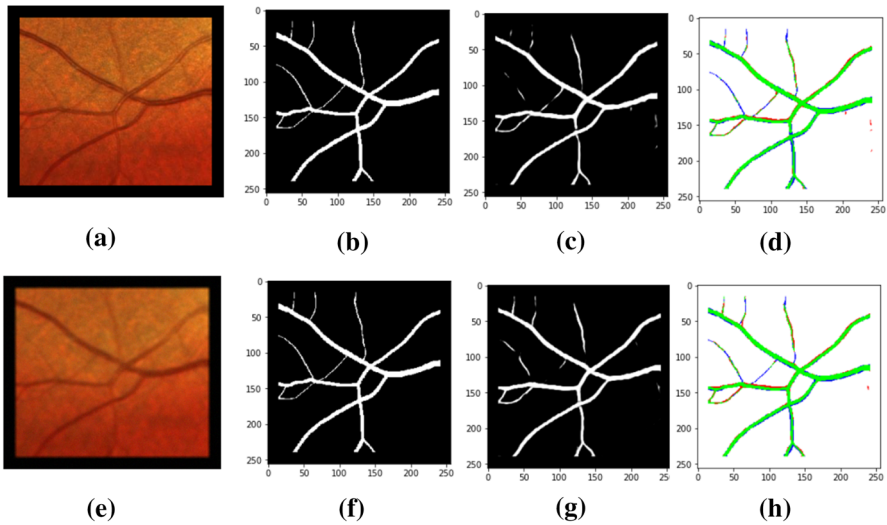


Fig. 21 Comparison of segmentation results for a noisy image with original image from RC-SLO database: **a** original color image, **b** corresponding ground truth image, **c** corresponding generator output image, **d** corresponding visual difference image, **e** Noisy image with $\sigma_b = 3$ and $\sigma_n = 0.02$, **f** corresponding ground truth image, **g** corresponding generator output image, and **h** corresponding visual difference image

larly, Fig. 21e represents a noisy color image (with $\sigma_b = 3$ and $\sigma_n = 0.02$) generated from original color image from RC-SLO database. Figure 21f–h represents the corresponding ground truth image, generator output image, and visual difference image, respectively. Visual difference image of the segmented blood vessels and ground truth for the image in RC-SLO database are shown in Fig. 21d, h, respectively. These two images are found to be identical, which represents the robustness of the proposed method.

6 Computational Complexity

The computational time for the blood vessel segmentation algorithm for the two types of discriminator determines the swiftness of the algorithm. The computational time of the proposed model mainly depends on the architecture and nature of the discriminator as well as the dimension of the database. Table 7 represents the training time and inference time for various fundus databases (DRIVE, STARE, CHASE_DB1, HRF, ARIA, IOSTAR, and RC-SLO databases) for the segmentation algorithm with VT and CNN-based classifier discriminators. The segmentation algorithm is trained by 105 images with 100 epochs. The training time is found to be highest for HRF database and least for RC-SLO database. Among the two different discriminators the CNN-based classifier is faster than vision transformer. This occurs due to complex architecture of vision transformer.

Table 7 Comparison of computational time for different databases using two types of discriminators

Algorithm	Dataset for training	Number of training images	Training time in sec	Inference time in sec/image
Vision transformer as discriminator	DRIVE	105	2044.57	0.0276
	STARE	105	1877.54	0.0287
	CHASE_DB1	105	1875.84	0.0286
	HRF	105	2442.45	0.0293
	ARIA	105	1874.40	0.0289
	IOSTAR	105	2218.38	0.0290
	RC-SLO	105	1857.55	0.0277
	DRIVE	105	1397.96	0.0271
CNN-based binary classifier as discriminator	STARE	105	1382.36	0.0278
	CHASE_DB1	105	1456.93	0.0285
	HRF	105	1805.15	0.0291
	ARIA	105	1465.28	0.0287
	IOSTAR	105	1615.91	0.0282
	RC-SLO	105	1289.65	0.0270

7 Conclusion

In this paper, retinal blood vessels are segmented from the color fundus images by utilizing multi-scale residual convolutional neural network (MSR-Net) combined with GAN. The generator in the GAN architecture utilizes deep residual blocks with skip connections in UNet backbone (MSR-Net) for segmentation, whereas the discriminator utilizes vision transformer for binary classification. This method was tested on various publicly available databases (DRIVE, STARE, CHASE_DB1, HRF, ARIA, IOSTAR, and RC-SLO). Various performance measures such as accuracy, precision, specificity, *AUC*, and *IoU* are calculated for analysis. The proposed method attains an accuracy of 0.9873 for CHASE_DB1 database, 0.9742 for DRIVE database, 0.9773 for HRF database, and 0.9628 for ARIA database. The proposed method is compared with recent state-of-the-art methods available in the literature. From the comparative analysis, it is found that the proposed blood vessel segmentation method outperforms existing methods used for vessel segmentation. Also, the proposed method is proved to be robust in the presence of noises affected to retinal images. The use of vision transformer as a discriminator increases the computational burden compared to the traditional binary classifier. The performance of this technique can be improved by implementing various GAN models like perceptual GAN and cyclic GAN. This may overcome the problems due to overfitting and may reduce the computational time.

Acknowledgements This work is carried out by the authors at National Institute of Technology Puducherry, Karaikal, India. IIT Guwahati (India) has supported for retinal database.

Data availability The authors confirm that the data supporting the findings of this study are available from the corresponding author upon request. The programs and the supporting files will be provided on request.

Declarations

Conflict of interest There is no conflicts of interest.

Ethical approval Our paper satisfies and fulfills the compliance with ethical standards. The research does not involve human participants and/or animals.

References

1. M.D. Abramoff, M.K. Garvin, M. Sonka, Retinal imaging and image analysis. *IEEE Rev. Biomed. Eng.* **3**, 169–208 (2010)
2. H. Akil, A.S. Huang, B.A. Francis, S.R. Sadda, V. Chopra, Retinal vessel density from optical coherence tomography angiography to differentiate early glaucoma, pre-perimetric glaucoma and normal eyes. *PLoS One* (2017)
3. J. Ba, J.R. Kiros, G.E. Hinton, Layer normalization. [arXiv:1607.06450](https://arxiv.org/abs/1607.06450) (2016)
4. K. Beom, S.H. Choi, J.Y. Lee, M-GAN: Retinal blood vessel segmentation by balancing losses through stacked deep fully convolutional networks. in *IEEE Access* (2020)
5. A. Budai, R. Bock, A. Maier, J. Hornegger, G. Michelson, Robust vessel segmentation in fundus images. *Int. J. Biomed. Imaging* (2013)
6. D. Chen, W. Yang, L. Wang, S. Tan, J. Lin, W. Bu, PCAT-UNet: UNet-like network fused convolution and transformer for retinal vessel segmentation. *PLOS ONE* (2022)
7. L. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-Decoder with atrous separable convolution for semantic image segmentation (2018). [arXiv:1802.02611](https://arxiv.org/abs/1802.02611)
8. B. Dashtbozorg, J. Zhang, F. Huang, B.M. ter Haar Romeny, Retinal microaneurysms detection using local convergence index features. *IEEE Trans. Image Process.* **27**(7), 3300–3315 (2018). <https://doi.org/10.1109/TIP.2018.2815345>
9. X. Deng, J. Ye, A retinal blood vessel segmentation based on improved D-MNet and pulse-coupled neural network. *Biomedical Signal Processing and Control* **73**, 103467 (2022). <https://doi.org/10.1016/j.bspc.2021.103467>
10. A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby: An image is worth 16 × 16 words: transformers for image recognition at scale (2020). [arXiv:2010.11929](https://arxiv.org/abs/2010.11929)
11. P. Elangovan, M.K. Nath, Glaucoma assessment from color fundus images using convolutional neural network. *Int. J. Imaging Syst. Technol.* (2020). <https://doi.org/10.1002/ima.22494>
12. H. Emami, M. Dong, S. Nejad-Davarani, C. Glide-Hurst, Generating synthetic CTs from magnetic resonance images using generative adversarial networks. *Med. Phys.* (2018). <https://doi.org/10.1002/mp.13047>
13. D. Farnell, F. Hatfield, P. Knox, M. Reakes, S. Spencer, D. Parry, S. Harding, Enhancement of blood vessels in digital fundus photographs via the application of multi-scale line operators. *J. Frankl. Inst.* **345**(7), 748–765 (2008). <https://doi.org/10.1016/j.jfranklin.2008.04.009>
14. M.M. Fraz, P. Remagnino, A. Hoppe, B. Uyyanonvara, A.R. Rudnicka, C.G. Owen, S.A. Barman, An ensemble classification-based approach applied to retinal blood vessel segmentation. *IEEE Trans. Biomed. Eng.* (2012)
15. H. Fu, Y. Xu, S. Lin, D.W.K. Wong, J. Liu, Deepvessel: retinal vessel segmentation via deep learning and conditional random field. In: *International conference on medical image computing and computer-assisted intervention* (2016)
16. I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks (2014)
17. Z. Gu, J. Cheng, H. Fu, K. Zhou, H. Hao, Y. Zhao, T. Zhang, S. Gao, J. Liu, CE-Net: context encoder network for 2D medical image segmentation. *IEEE Trans. Med. Imaging* (2019)
18. X. Guo, C. Chen, Y. Lu, K. Meng, H. Chen, K. Zhou, Z. Wang, R. Xiao, Retinal vessel segmentation combined with generative adversarial networks and dense UNet. *IEEE Access* (2020)
19. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016)

20. J. Hoffman, E. Tzeng, T. Park, J.Y. Zhu, P. Isola, K. Saenko, A. Efros, T. Darrell, Cycada: cycle-consistent adversarial domain adaptation. in *International Conference on Machine Learning*, pp. 1989–1998 (2018)
21. A. Hoover, V. Kouznetsova, M. Goldbaum, Locating blood vessels in retinal images by piece-wise threshold probing of a matched filter response. *IEEE Trans. Med. Imaging*, pp. 203–210 (2000)
22. K. Hu, Z. Zhang, X. Niu, Y. Zhang, C. Cao, F. Xiao, X. Gao, Retinal vessel segmentation of color fundus images using multi-scale convolutional neural network with an improved cross-entropy loss function. *J. Neurocomput.* (2018)
23. Q. Jin, Z. Meng, T.D. Pham, Q. Chen, L. Wei, R. Su, DUNet: a deformable network for retinal vessel segmentation. *Knowl.-Based Syst.* (2019)
24. M. Kar, M. Nath, M. Mishra, Retinal vessel segmentation and disc detection from color fundus images using inception module and residual connection. in *3rd International Conference On Recent Trends In Advanced Computing* (2020)
25. J. Kohler, A. Budai, M.F. Kraus, J. Odstrcilik, G. Michelson, J. Hornegger, Automatic no-reference quality assessment for retinal fundus images using vessel segmentation. in *Proceedings of the 26th IEEE International Symposium on Computer-based Medical Systems* (2013)
26. J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, S. Yan, Perceptual generative adversarial networks for small object detection (2017). [arXiv:1706.05274](https://arxiv.org/abs/1706.05274)
27. X. Li, Z. Du, Y. Huang, Z. Tan, A deep translation (GAN) based change detection network for optical and SAR remote sensing images. *ISPRS J. Photogrammetry Remote Sens.* **179**, 14–34 (2021). <https://doi.org/10.1016/j.isprsjprs.2021.07.007>
28. S. Moccia, E.D. Momi, S.E. Hadji, L.S. Mattos, Blood vessel segmentation algorithms: review of methods, datasets and evaluation metrics. *Comput. Methods Programs Biomed.* (2018)
29. M. Niemeijer, B. Ginneken, M. Loog, Comparative study of retinal vessel segmentation methods on a new publicly available database, in *Proceedings of SPIE - The International Society for Optical Engineering*, (2004)
30. C.G. Owen, A.R. Rudnicka, R. Mullen, S.A. Barman, D.N. Monekosso, P.H. Whincup, J. Ng, C. Paterson, Measuring retinal vessel tortuosity in 10-year-old children: validation of the computer-assisted image analysis of the retina (CAIAR) program. *Investig. Ophthalmol. Visual Sci.* **50**(5), 2004–10 (2009)
31. K.B. Park, S.H. Choi, J.Y. Lee, M-GAN: Retinal blood vessel segmentation by balancing losses through stacked deep fully convolutional networks. in *IEEE Access* (2020)
32. S.M. Pizer, E.P. Amburn, J.D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B.M. ter Haar Romeny, J.B. Zimmerman, Adaptive histogram equalization and its variations. *Comput. Vis. Graphics. Image Process.* **39**, 355–368 (1987)
33. O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation. in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 1–8 (2017)
34. K. Santosh, S. Ghosh, M. Bose, Ret-GAN: Retinal image enhancement using generative adversarial networks, pp. 79–84 (2021). <https://doi.org/10.1109/CBMS52027.2021.00082>
35. R.A. Shehhi, P.R. Marpu, W.L. Woon, An automatic cognitive graph-based segmentation for detection of blood vessels in retinal images. *Math. Probl. Eng.* (2016). <https://doi.org/10.1155/2016/7906165>
36. S.Y. Shin, S. Lee, I.D. Yun, K.M. Lee, Deep vessel segmentation by learning graphical connectivity. *Med. Image Anal.* **58**, 1–14 (2019)
37. J. Son, S.J. Park, K.H. Jung, Retinal vessel segmentation in fundoscopic images with generative adversarial networks. [arXiv:1706.09318](https://arxiv.org/abs/1706.09318) (2017)
38. E. Uysal, G.E. Guraksin, Computer-aided retinal vessel segmentation in retinal images: convolutional neural networks. *Multimedia Tools Appl.*, pp. 1929–1958 (2020)
39. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need. (2017). [arXiv:1706.03762](https://arxiv.org/abs/1706.03762)
40. D. Vijayalakshmi, M.K. Nath, O.P. Acharya, A comprehensive survey on image contrast enhancement techniques in spatial domain. *Sens. Imaging* **21** (2020). <https://doi.org/10.1007/s11220-020-00305-3>
41. Y. Xue, T. Xu, H. Zhang, L.R. Long, X. Huang, SegAN: Adversarial network with multi-scale L1 loss for medical image segmentation. *Neuroinformatics*, pp. 383–392 (2018)
42. Z. Yan, X. Yang, K.T. Cheng, Joint segment-level and pixel-wise losses for deep learning based retinal vessel segmentation. *IEEE Trans. Biomed. Eng.*, pp. 1912–1923 (2018)

43. X. Yu, X. Cai, Z. Ying, T.H. Li, G. Li, SingleGAN: image-to-Image translation by a single-generator network using multiple generative adversarial learning. (2018). [arXiv:1810.04991](https://arxiv.org/abs/1810.04991)
44. H. Zhang, I. Goodfellow, D. Metaxas, A. Odena, Self-attention generative adversarial networks (2019)
45. J. Zhang, B. Dashtbozorg, E. Bekkers, J.P.W. Pluim, R. Duits, B.M.T.H. Romeny, Retinal vessel segmentation via locally adaptive derivative frames in orientation scores. *IEEE Trans Med Imaging* (2016)
46. Y. Zhang, M. He, Z. Chen, K. Hu, X. Li, X. Gao, Bridge-Net: context-involved UNet with patch-based loss weight mapping for retinal blood vessel segmentation. *Expert Syst. Appl.* **195**, 116526 (2022) <https://doi.org/10.1016/j.eswa.2022.116526>
47. Y. Zhao, Y. Liu, X. Wu, S.P. Harding, Y. Zheng, Retinal vessel segmentation: an efficient graph cut approach with retinex and local phase. *PLoS ONE* **10**(4), 1–22 (2015). <https://doi.org/10.1371/journal.pone.0122332>
48. Z. Zhou, M.M.R. Siddiquee, N. Tajbakhsh, J. Liang, UNet++: redesigning skip connections to exploit multiscale features in image segmentation. *IEEE Trans Med Imaging*. (2020)
49. J. Zhu, T. Park, P. Isola, A.A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks (2017). [arXiv:1703.10593](https://arxiv.org/abs/1703.10593)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.