D-GaussianNet: Adaptive Distorted Gaussian Matched Filter with Convolutional Neural Network for Retinal Vessel Segmentation

Dora E. Alvarado-Carrillo1, Emmanuel Ovalle-Magallanes2, and Oscar S. Dalmau-Cedeño1

1 Center for Research in Mathematics (CIMAT), 36000 Guanajuato, GTO, Mexico
{dora.alvarado,dalmau}@cimat.mx
2 University of Guanajuato, 36885 Salamanca, GTO, Mexico
e.ovallemagallanes@ugto.mx

Abstract. Automating retinal vessel segmentation is a primary element of computer-aided diagnostic systems for many retinal diseases. It facilitates the inspection of shape, width, tortuosity, and other blood vessel characteristics. In this paper, a new method that incorporates Distorted Gaussian Matched Filters (D-GMFs) with adaptive parameters as part of a Deep Convolutional Architecture is proposed. The D-GaussianNet includes D-GMF units, a variant of the Gaussian Matched Filter that considers curvature, placed at the beginning and end of the network to implicitly indicate that spatial attention should focus on curvilinear structures in the image. Experimental results on datasets DRIVE, STARE, and CHASE show state-of-the-art performance with an accuracy of 0.9565, 0.9647, and 0.9609 and a F1-score of 0.8233, 0.8141, and 0.8077, respectively.

Keywords: Retinal blood vessel segmentation · Gaussian matched filter · Convolutional neural network

1 Introduction

Fundus retinal imaging is a popular non-invasive technique for monitoring the retinal structure, which consists of obtaining a two-dimensional projection of the retinal semitransparent tissues, using specialized cameras with reflective light [1,5]. Fundus images are highly important for the early diagnosis and tracking of various diseases related to vascular changes, such as diabetic retinopathy, age-related macular degeneration, and glaucoma, since the retina is the only structure that allows direct imaging of blood circulation [3,22,37].

This work is partially supported by CONACyT, Mexico (Doctoral Studies Grants no. 626155 and 626154).

© Springer Nature Switzerland AG 2021
M. Nguyen et al. (Eds.): ISGV 2021, CCIS 1386, pp. 378–392, 2021.
https://doi.org/10.1007/978-3-030-72073-5_29
The retinal vascular tree is a major structure to be studied in the analysis of fundus images. Inspection of shape, width, tortuosity, and other blood vessel characteristics contributes to identify many retinal diseases [4,18]. Moreover, the detection and subtraction of the vascular tree facilitate the recognition of other lesions that appear in the retina, so a precise delineation of the vessels is required. However, fundus images are characterized by low contrast and notable illumination changes [32,41]. In addition, blood vessels consist of a varied morphology: vessels are bifurcated, tortuous, and their width is reduced to extremely thin sections. Hence, blood vessel detection is a time-consuming task, its manual fulfillment is usually limited by the availability of ophthalmologists in the healthcare system, resulting in diagnosis delays and elevated treatment costs [5,19].

As an alternative to contribute to the early diagnosis of eye diseases, several methods for automating blood vessel segmentation have been presented in recent years. Nevertheless, the task is considered a challenging problem up to date, since there are conditions—for instance, the similarity in color between vessels and microaneurysms, the central reflex in some vessels, the presence of multiple branching points and neovascularization, among others—where the state-of-the-art algorithms have difficulties to perform an accurate result. Furthermore, many recent deep learning methods use generic architectures for feature extraction, whose performance is strongly tied to the quantity and quality of the examples available in the training phase. Since in the area of biomedical image processing, it is common to work with a limited number of examples, different training strategies are required. Frequently, this issue is overcome by applying transferred learning [18,25,29] or using patches rather than full images for training [9,11,19,23], while a few papers have proposed hybrid techniques for feature extraction [12,34]. In this paper, a new method that incorporates distorted Gaussian Matched Filters with adaptive parameters as part of a Deep Convolutional Architecture is proposed. The strategy aims to alleviate the traditional neural convolutional models’ dependence on large datasets, by aggregating robustness through a hybrid design that considers both a priori information about the curvilinear shape of the vessels, as well as deep learning techniques. The contributions of this paper are as follows: first, a new technique for incorporating curvature into Gaussian Matched Filters, Distorted GMF (D-GMF), is presented. The approach intends to relax the strong assumption that blood vessels are piecewise linear; second, a preprocessing method based on random quantum circuits is integrated into a deep learning strategy for blood vessel segmentation. To the best of the authors’ knowledge, this is the first time a quantum preprocessing has been applied for fundus images, although there are few related works in the area of biomedical imaging [2,14]; third, a novel neural convolutional architecture with adaptive units of distorted Gaussian Matched Filters is presented, the units are placed at the beginning and end of the network to implicitly indicate that spatial attention should focus on elongated structures in the image.
The rest of the paper is organized as follows. In Sect. 2, a brief review of the related work is presented. In Sect. 3, the key elements of the proposed method—quantum preprocessing, distorted Gaussian Matched Filters, and the Convolutional Neural architecture—are explained in detail. In Sect. 4, experimental results and comparison with state-of-the-art approaches are presented. In Sect. 5, a discussion on the proposed method is carried out.

2 Related Work

Based on the type of information applied in the process, automatic blood vessel segmentation methods can be classified into two large groups: unsupervised and supervised methods. For both approaches, a brief review of related work is presented below.

2.1 Unsupervised Methods

Unsupervised methods purely use prior knowledge about vascular structure—such as length, width, and grayscale intensity—to design strategies that highlight the vessels.

Template filtering methods use predefined templates to model the elongated shape of the vessels. Khan et al. [20] present generalized multiscale line detectors, by proposing adjustable window size in filters: the window and line sizes grow proportionally, instead of fixing one or both of them. Filter orientations are estimated locally. Line detectors perform well for detecting thin vessels, as the line template always maintains a width of one pixel. However, the resulting images often contain artifacts and require post-processing.

Morphological methods use operations such as dilation and erosion to discard round structures and highlight curvilinear vessel alike elements. Sazak et al. [28] propose a method based on two banks of structuring elements at different scales. The first bank contains disks with varied diameters, the second bank contains lines of different thickness and orientation. For both banks, opening operations are performed with the input image. Then, difference images are computed between disk and line results, the maximum response is preserved. This combination of structuring elements produces better segmentation results in vessel bifurcations. Pal et al. [26] use morphological operations to improve image contrast and segment the vascular structure. First, a Top-Hat Transform is combined with the wavelet Transform to highlight the vessels, improve contrast, and eliminate the optic disc; then, the Hit-or-Miss Transform is applied with two structuring elements, one to highlight the background and the other to highlight vessels at different scales. For each scale, the difference between the two outputs is obtained, the final image is computed as the maximum of these differences.

Tracking methods use vessel connectivity to distinguish the vascular region. Given an initial point and direction, the vessels’ edges and their bifurcations are determined by exhaustive search using tracking strategies. Zhao et al. [40] propose an algorithm based on multiscale SLIC superpixels to reduce the number
of possible paths to be traced. The multiscale is built by varying the superpixel size, then a filtering process is implemented to select the correct scale, considering intensity and variance.

The methods reviewed in this section do not need additional information to perform the segmentation task. However, their performance is moderate, as their ability to model the intricate vascular structure is limited, noise artifacts are created in the output image, and require meticulous selection of their parameters.

### 2.2 Supervised Methods

Supervised methods require manually labeled data. The segmentation task is approached as a classification problem: a classifier is trained—using labeled examples—to assign a class to each pixel in the input image, so that the pixels with the same label share certain features. In the case of blood vessel segmentation, the classification is binary, since the goal is to differentiate the region with blood vessels, i.e., class 1, from the rest of the image, i.e., class 0. Condurache et al. [7] propose to use different filters (Bothat transform, Hessian Matrix, Laplacian Pyramid) to obtain vascular maps that emphasize pixels belonging to elongated structures of a certain width. From these maps, feature vectors are created for each pixel, which feeds a classifier based on hysteresis. Wang et al. [36] present a different method with a divide and conquer strategy, which consists of clustering pixels according to the width of the blood vessel and its position in it (whether it is in the center or on the edge). The clusters are obtained using 2-D Gabor wavelet kernels, then a funnel architecture composed of SVM classifiers is used to classify the pixels, initially in three classes (vessel, no vessel, uncertain) and in a final stage in two classes. Liskowski et al. [23] propose a neural network for automatic blood vessel segmentation. The architecture consists of four convolutional layers with an increasing number of filters (64, 64, 128, and 128 filters, respectively) and three fully connected layers (521, 512, and 2 neurons, respectively). Feng et al. [9], inspired by the design of the UNet [27], propose a cross-architecture in which the most superficial layers are densely connected to the deepest layers of the network. The model does not use an encoder-decoder design, instead each block outputs an activation map of the same dimensions, which simplifies the fusion between maps from different levels in the network. Jin et al. [19] propose to replace traditional convolutions with deformable convolutions to overcome some of the limitations of the UNet model. Adaptive receptive fields are included in the network’s design to capture the complicated structure of blood vessels. The architecture shows the ability to extract thin vessels, but the computational time increases considerably, compared to models on which it is based, UNet and Deformable-ConvNet. Li et al. [21] propose to take as a starting point an initial prediction obtained with a UNet architecture and to refine the result iteratively, using a simplified version of the UNet, called mini-unet. This approach allows the model to make corrections on the initial prediction, e.g., disconnected vessel segments that should be connected. The number of iterations of the mini-unet is a parameter that must be adjusted carefully, as a bad selection may cause overfitting.
Manual feature extraction methods have a direct interpretation of the criteria used for segmentation and their performance is good when trained with a database of moderate size. However, in recent years these techniques have been surpassed by the ability of Deep Neural Networks for automatic feature extraction and model generalization. This last approach also has its limitations, it requires a greater number of examples in the training step, and like its hand-crafted counterpart has difficulties detecting thin vessels, bifurcations, and central vessel regions with reflexes, among other cases.

In this paper, a new method that considers both a priori information on the vessels’ geometry and automatic extraction of features is presented. The following section describes the algorithm in detail.

3 Methods/Methodology

The proposed method consists of a Residual Network where Distorted Gaussian Matched Filters have been incorporated. In this section, we briefly cover the topics of Quantum Convolutional Layers for Image Preprocessing, Residual Networks, and Gaussian Matched Filters, since these elements are a fundamental part of the algorithm. Subsequently, the proposed strategy to incorporate a curvature component to GMFs and their incorporation into the neuronal architecture are described in detail.

3.1 Quantum Preprocessing

Fundus images are affected by illumination, producing variations in background intensity. Some preprocessing techniques can be applied to improve the contrast of medical images, such as Contrast Limited Adaptive Histogram Equalization (CLAHE). In this work, a new type of preprocessing by leveraging certain promising quantum computation aspects was investigated. A so-called Quantum Convolutional Layer (QCL) [15] was applied to generate a multi-channel fundus image. Given a neighborhood $\Omega_x$ of size $n \times n$ where $n > 1$, around a pixel position $(u, v)$, an encoding function $e$, quantum circuit parameters $q$ (e.g., a set of Pauli Gates), and a decoding measurement $d$, a Quantum Convolutional Layer transform the input data into different output channel pixel values at position $(u, v)$, such as:

$$f_x = Q(\Omega_x, e, q, d), \quad f_x : \mathbb{R}^{n \times n \times 1} \rightarrow \mathbb{R}^{1 \times 1 \times (n \cdot n)}.$$  (1)

Furthermore, unlike the classical convolution, which convolves a filter through the image, a QCL transforms input data employing a quantum circuit. Figure 1 shows and example of a quantum circuit applied for each neighborhood $\Omega_x$.

3.2 The Distorted Gaussian Matched Filter (D-GMF)

For this part, a new technique to incorporate curvature to the classic Gaussian Matched Filter (GMF) is presented.
The Gaussian Matched Filter (GMF). The GMF is a method for highlighting tubular structures in two-dimensional images [6]. It is motivated by the observation that a blood vessel’s grayscale intensity profile resembles a one-dimensional Gaussian distribution. Under the assumption that vessels are piecewise linear structures, the GMF can be used to simultaneously identify vessel sections with the same amplitude. It is defined as follows:

\[ k(x, y) = -\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2\sigma^2}\right), \quad \forall |x| \leq 3\sigma, \quad \forall |y| \leq \frac{L}{2}, \quad (2) \]

where \( \sigma \) is a parameter that controls the amplitude of the Gaussian profile, and its value is selected according to the width in pixels of the structures to be highlighted. \( L \) represents the length for which said structures are assumed to be linear, i.e., it corresponds to the filter’s size.

Multiple variants of the original GMF have been proposed, not only in the medical area, but in any field where the detection of curvilinear objects is required [8, 24, 35, 38]. However, GMF and its variants are based on a strong assumption about curvature, claiming that for an appropriate value of \( L \) curvilinear objects do not present a significant level of curvature and can be considered linear. Thus, parameters \( L \) and \( \sigma \) become fundamental for the method’s good performance.

The Distorted Gaussian Matched Filter (D-GMF). The D-GMF is inspired by techniques used to generate additional data in handwriting recognition: besides the distortions that can be applied to any image (such as translations, rotations, zooming and skewing), handwritten characters can also have variations related to physical factors such as hand movement, and which can be modeled through elastic deformations [30]. Analogously, blood vessels and other curvilinear structures are not completely straight but can have a degree of curvature that makes them difficult to detect with the original GMFs. To model this curvature, an elastic deformation transformation is applied to the original GMF filter.
Elastic deformations [30] can be modeled by generating random displacement fields, $\Delta x$ and $\Delta y$ for horizontal and vertical directions, respectively. First, random fields $\text{Rand}_x$ and $\text{Rand}_y$ are generated with uniform distribution in the interval $[-1,1]$. Then, a two-dimensional Gaussian filter with a standard deviation of $\beta$, $G(\beta)$, is convolved with said fields to ensure similar displacements around a neighborhood. Finally, a scale factor $\alpha$ is applied to obtain the final displacement fields, as shown in Eq. (3) and (4).

$$\Delta x = \alpha \cdot \left( \text{Rand}_x(m, n) \ast G(\beta) \right),$$  \hspace{1cm} (3)  

$$\Delta y = \alpha \cdot \left( \text{Rand}_y(m, n) \ast G(\beta) \right),$$  \hspace{1cm} (4)

where $\ast$ denotes the operand of discrete convolution, and $m$ and $n$ are the displacement field’s height and width, respectively. Then, each pixel with position $(x, y)$ in the GMF filter, i.e., $k(x, y)$, is mapped to a new position $(u, v)$, determined by its displacement components $\Delta x(x, y)$ and $\Delta y(x, y)$, as specified in the following equation:

$$\tilde{k}(u, v) = k(x + \Delta x(x, y), y + \Delta y(x, y))$$  \hspace{1cm} (5)

The Gaussian filter $G(\beta)$ locally averages the uniform random fields: if $\beta$ is very small, then $\text{Rand}_x$ and $\text{Rand}_y$ are practically not affected by the filter and keep their random appearance; In contrast, if $\beta$ is very large, the displacements are very small (their average being close to zero), and the deformations are imperceptible; Intermediate values in the range $[4,8]$ are recommended to obtain the appearance of elastic deformation. The value of $\alpha$ determines the intensity of the distortion, i.e., the curvature level in the Gaussian Matched Filter.

Figure 2 illustrates a GMF filter and its distorted versions using uniform random displacements and elastic deformations.

![Fig. 2](image)

**Fig. 2.** (a) Original GMF filter with parameters $(\sigma = 1$, $L = 21)$, (b) Distorted GMF filter using uniform random displacements in the interval $(-1,1)$, (c) Distorted GMF filter using elastic deformations with parameters $(\beta = 4$, $\alpha = 30)$. The green arrows represent the pixel displacement vectors.

By integrating a curvature level, the Distorted Gaussian Matched Filter (D-GMF) aims to overcome some limitations that classical GMFs have in modeling
complicated shapes. In the following sections, an end-to-end convolutional neural network incorporating D-GMF filters is presented, so that amplitude and curvature can be automatically adjusted in the training phase while guiding the network’s spatial attention to highlight blood vessels—or curvilinear elements in general—of the input images.

3.3 Adaptive Distorted Gaussian Matched Filters with Convolutional Neural Network

The overall D-GaussianNet architecture is presented in Fig. 3. The structure is based on the UNet [27], formed by three main parts—encoder, bottleneck, and decoder—and skip-connections between feature maps from the encoder path to the decoder path. The model proposed in this study differs from the original UNet in various aspects. First, stem blocks—formed by a D-GMF layer and two convolutional layers—have been added at the beginning and end of the network, in order to implicitly indicate that the spatial attention of the network must be focus on the curvilinear structures of the image. The D-GMFs have not been placed in levels with less dimensionality than the original image to maintain the interpretation, so the input map has the same dimensions of the original image and its spatial characteristics are mostly preserved, therefore it seems reasonable to apply the proposed filters.

Besides, the conventional convolutional blocks of each level have been replaced with residual convolutional blocks, similarly to previous works presented for segmentation of curvilinear objects in [17,39].

This modification is motivated by the nature of the model, which incorporates D-GMF blocks to perform adaptive curvilinear structures enhancement on the same neuronal architecture. This could lead to problems such as premature convergence or degradation of training accuracy. However, residual models alleviate this problem by proposing a mechanism that includes a reformulation of the convolutional block mapping function, so that they also learn a residual component between the desired transformation and the input of the block. Residual networks are concisely explained in the following paragraphs.

The Residual Model. A Residual Neural Network [13], or ResNet, is an architecture that simulates identity mapping functions inside its blocks of convolutional layers, by using shortcut connections that skip intermediate layers and directly add the input feature map of each block to its last layer.

A difficulty encountered by Deep Neural Networks is the degradation of training accuracy, which occurs when—by increasing the depth of a network—accuracy becomes saturated and degrades rapidly.

The ResNet focuses on solving this problem by explicitly posing a residual mapping layer. For a certain block of layers in the network, let \( x \) be the input feature map, \( \{W_i\} \) the weights of the \( i \)-th layer and \( \mathcal{H}(x) \) the desired mapping to learn, the residual between both is given by the expression:

\[
\mathcal{F}(x, \{W_i\}) = \mathcal{H}(x) - x.
\]
Assuming that it is easier to optimize the residual mapping $F(x)$ than the original $H(x)$—since intuitively it is simpler for a layer to learn a mapping to zero than to learn the identity function, it is proposed to reorder the above equation as:

$$H(x) = F(x, \{W_i\}) + x. \quad (7)$$

This change preserves the main objective of learning the desired mapping $H(x)$, with the difference that at the same time, the network also learns the residual $F(x, \{W_i\})$.

The residual Eq. (7) is achieved by incorporating shortcut connections that link the input feature map of a block with its output, so that both can be added. Taking advantage of this design, a Residual Architecture with adaptive D-GMFs is proposed in this study. The residual mechanism proposed for the adaptive incorporation of D-GMFs over a Deep Neural Network is described.

**The Adaptive D-GMF Unit.** The D-GMF Adaptive Unit consists of a convolutional layer of D-GMF filters, followed by two conventional convolutional layers, between which Batch-Normalization and ReLu-Activation are applied. The block incorporates a residual connection that connects the input of the
block with the second convolution output in an Addition layer, where both feature maps are merged.

The D-GMF convolutional layer only requires the configuration of a number of filters $N$, while the amplitudes and curvature levels of each filter are automatically adjusted in the training phase of the neural network. The behavior of the layer is as follows, let $\{\sigma_j\}$ and $\{\alpha_j\}$ be two parameters sets that specify amplitude and curvature level of $N$ filters in a D-GMF layer, with $1 \leq j \leq N$.

1. Filter parameters $\{\sigma_j\}$ and $\{\alpha_j\}$ are randomly initialized,
   a. Values in $\{\sigma_j\}$ are initialized using a continuous uniform distribution in the interval $(a, b)$, i.e., $\sigma_j \sim U(a, b)$,
   b. Values in $\{\alpha_j\}$ are initialized using a continuous uniform distribution in the interval $(c, d)$, i.e., $\alpha_j \sim U(c, d)$.
   where tuple $(a, b)$ corresponds to the lower and upper boundaries for the initial amplitude values, and tuple $(c, d)$ corresponds to the lower and upper values for the initial curvature values.

2. For each filter $f_j$,
   a. a set $\{f_j(\theta)\}$ composed of rotated versions of $f_j$ is constructed, where $\theta$ is the rotation angle, which takes evenly spaced values in the interval $[0, 2\pi]$;
   b. the filters in set $\{f_j(\theta)\}$ are applied to the layer’s input feature map $I$, obtaining responses $\{R_j(\theta)\}$;
   c. the filter’s output $O_j$ is computed by keeping the maximum response at element level among the set of responses $\{R_j(\theta)\}$, i.e.,
      \[
      O_j(x, y) = \max\{R_j(\theta)(x, y)\}
      \]
      where $(x, y)$ is a position in the input feature map domain.
   d. The set of filter outputs $\{O_j\}$ constitutes the output feature map of the layer.

3.4 Datasets

The proposed model was evaluated with three public datasets: DRIVE, STARE and CHASE. The dataset DRIVE [33] consists of 40 images of $565 \times 584$ pixels, which are divided into a training set and a test set, of 20 images each; There are two sets of manually segmented images available, created by observers instructed by an expert. The dataset STARE [16] consist of 20 images of $700 \times 605$ pixels, which were manually labeled to produce ground-truth vessel segmentation images; The set contains 10 images without pathologies and 10 images with pathologies that obscure the image. The dataset CHASE [10] consist of 28 images of $999 \times 960$ pixels, acquired from 14 children from multi-ethnic schools in England; the images were manually labeled by two different observers. For STARE and CHASE datasets, training and test partitions are not provided, therefore sets of 10 and 14 images were used for training, and sets of 10 and 14 images for testing, respectively.
3.5 Performance Evaluation Metrics

The metrics considered for the evaluation of the method are Sensitivity (TPR), Specificity (TNR), Precision (PPV), Accuracy (ACC) and F1-Score (F1).

TPR refers to the proportion of pixels that are correctly classified as positive, among the total number of positive pixels. TNR measures the proportion of pixels that are correctly classified as negative, among the total number of negative pixels. PPV refers to the proportion of pixels that are correctly classified as positive, among the total number of pixels examined. ACC indicates the proportion of pixels correctly classified, both positives and negatives, among the total number of pixels examined. The F1 score is the harmonic average of the PPV and the TPR. These metrics have the following expressions.

\[
\text{TPR} = \frac{TP}{TP + FN} \quad (9)
\]

\[
\text{TNR} = \frac{TP}{TP + FN} \quad (10)
\]

\[
\text{PPV} = \frac{TP}{TP + FP} \quad (11)
\]

\[
\text{ACC} = \frac{TP + TN}{TP + FP + TN + FN} \quad (12)
\]

\[
\text{F1} = 2 \cdot \frac{\text{PPV} \cdot \text{TPR}}{\text{PPV} + \text{TPR}} \quad (13)
\]

Where TP represents the number of true positive pixels, TN represents the number of true negative pixels, FP represents the number false positive pixels and FN represents the number of false positive pixels.

Additionally, the performance was evaluated with the Area Under the ROC Curve (AUC), which refers to the area under a curve that allows evaluating the quality of a result by calculating the rate of change of Sensitivity (TPR) versus Specificity (TNR).

4 Experimental Results

The model was trained from scratch on random patches, extracted from the preprocessed images. As described in Sect. 3.4, each dataset was divided in training and test sets. Additionally, a 10% percentage of the training set was used as a validation set for the parameter selection. Only pixels within the Field of View (FOV) were considered in the evaluation. For the STARE and CHASE datasets, which do not provide a binary mask to identify the FOV, they were created using a manually chosen global threshold.

The patch size was modified during the training process, using a progressive resizing strategy as follows. For a total of 90 epochs, the training process was divided into three parts of 30 epoch each: first, a patch size of $24 \times 24$ pixels was used; then, a patch size of $32 \times 32$ pixels was used; finally, a patch size of $48 \times 48$ pixels was used. As the patch size changed, the best weights from
the previous part were used to initialize the model weights. For the evaluation, ordered patches extracted from the preprocessed images, of size $48 \times 48$ with an overlapping of 10 pixels were used. The optimization configuration consist of the Adam optimizer with a cyclical learning rate strategy as described in [31] with an initial learning rate set to 0.005. Binary cross-entropy was used as loss function. All experiments were implemented using the Pytorch framework and an NVIDIA Tesla K80 GPU.

The proposed method was compared with various state-of-the-art methods, including methods based on complete image prediction, patch-based prediction, and pixel-based prediction (from a patch neighborhood). The evaluation was performed using the metrics presented in Sect. 3.5. From the Table 1, it is shown that the proposed method is competitive with state-of-the-art methods: for CHASEDB and DRIVE databases, it obtains the highest and second highest score in the F1 metric, respectively; for STARE and DRIVE databases, it obtains the highest and second highest value in TPR, respectively, and the second-best value in ACC for both cases.

### Table 1. Performance comparison against state-of-the-art methods on DRIVE, STARE and CHASE datasets.

| Dataset    | Method/Author          | TPR   | TNR   | AUC   | ACC   | F1    |
|------------|------------------------|-------|-------|-------|-------|-------|
| DRIVE      | DNN/Liskowski et al. [23] | 0.7811 | 0.9807 | 0.9790 | 0.9535 | –     |
|            | CcNet/Feng et al. [9]   | 0.7625 | 0.9809 | 0.9678 | 0.9528 | –     |
|            | DUNet/Jin et al. [19]   | 0.7963 | 0.9800 | 0.9802 | 0.9566 | 0.8237 |
|            | IterNet/Li et al. [21]  | 0.7791 | 0.9831 | 0.9813 | 0.9574 | 0.8218 |
|            | Proposed method         | 0.7960 | 0.9799 | 0.9772 | 0.9565 | 0.8233 |
| STARE      | DNN/Liskowski et al. [23] | 0.7448 | 0.9828 | 0.9720 | 0.9525 | –     |
|            | CcNet/Feng et al. [9]   | 0.7709 | 0.9848 | 0.9700 | 0.9633 | –     |
|            | DUNet/Jin et al. [19]   | 0.7595 | 0.9878 | 0.9832 | 0.9641 | 0.8143 |
|            | IterNet/Li et al. [21]  | 0.7715 | 0.9886 | 0.9881 | 0.9701 | 0.8146 |
|            | Proposed method         | 0.7904 | 0.9843 | 0.9837 | 0.9647 | 0.8141 |
| CHASEDB    | DUNet/Jin et al. [19]   | 0.8155 | 0.9752 | 0.9804 | 0.9610 | 0.7883 |
|            | IterNet/Li et al. [21]  | 0.7970 | 0.9823 | 0.9851 | 0.9655 | 0.8073 |
|            | Proposed method         | 0.7530 | 0.9863 | 0.9798 | 0.9609 | 0.8077 |

### 5 Conclusions

In this paper, a novel end-to-end convolutional neural network for the automatic retinal vessel segmentation was proposed. Herein, a new variant of the Gaussian Matched Filters that incorporates curvature to the filter was presented. It improves the modeling of curvilinear structures, detecting tortuosity and other intricate shapes in vessels. The incorporation of distorted Gaussian Matched
filters on a residual convolutional architecture allowed the amplitude and curvature parameters to be adjusted automatically, based on the input images. Additionally, a Quantum Convolutional Layer was incorporated as a new type of preprocessing. The experimental results show that the proposed method has a competitive performance with the state-of-the-art methods, and even surpasses in terms of sensitivity to the state-of-the-art methods for the STARE database, and in terms of F1-score for the CHASE database.

References

1. Abramoff, M.D., Garvin, M.K., Sonka, M.: Retinal imaging and image analysis. IEEE Rev. Biomed. Eng. 3, 169–208 (2010)
2. Acar, E., Yilmaz, I.: Covid-19 detection on IBM quantum computer with classical-quantum transfer learning. medRxiv (2020)
3. Amil, P., Reyes-Manzano, C.F., Guzmán-Vargas, L., Sendiña-Nadal, I., Masoller, C.: Network-based features for retinal fundus vessel structure analysis. PloS one 14(7), e0220132 (2019)
4. Badawi, S.A., Fraz, M.M.: Optimizing the trainable B-COSFIRE filter for retinal blood vessel segmentation. PeerJ 6, e5855 (2018)
5. Chalakkala, R.J., Abdullaa, W.H., Hongb, S.C.: Fundus retinal image analyses for screening and diagnosing diabetic retinopathy, macular edema, and glaucoma disorders. Diabetes and Fundus OCT, p. 59 (2020)
6. Chaudhuri, S., Chatterjee, S., Katz, N., Nelson, M., Goldbaum, M.: Detection of blood vessels in retinal images using two-dimensional matched filters. IEEE Transactions on medical imaging 8(3), 263–269 (1989)
7. Condurache, A.P., Mertins, A.: Segmentation of retinal vessels with a hysteresis binary-classification paradigm. Comput. Medi. Imaging Graph. 36(4), 325–335 (2012)
8. Cruz-Aceves, I., Cervantes-Sanchez, F., Avila-Garcia, M.S.: A novel multiscale gaussian-matched filter using neural networks for the segmentation of x-ray coronary angiograms. J. Healthcare Eng. 2018 (2018). https://doi.org/10.1155/2018/5812059
9. Feng, S., Zhuo, Z., Pan, D., Tian, Q.: CcNet: a cross-connected convolutional network for segmenting retinal vessels using multi-scale features. Neurocomputing 392, 268–276 (2020)
10. Fraz, M.M., et al.: Ensemble classification system applied for retinal vessel segmentation on child images containing various vessel profiles. In: Campilho, A., Kamel, M. (eds.) Image Analysis and Recognition, pp. 380–389. Springer, Heidelberg (2012)
11. Fu, Q., Li, S., Wang, X.: MSCNN-AM:: a multi-scale convolutional neural network with attention mechanisms for retinal vessel segmentation. IEEE Access 8, 163926–163936 (2020)
12. Geng, L., Li, P., Zhu, W., Chen, X.: M2E-Net: multiscale morphological enhancement network for retinal vessel segmentation. In: Pen, Y., et al. (eds.) PRCV 2020, Part. LNCS, vol. 12305, pp. 493–502. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-60633-6_41
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
14. Heidari, S., Naseri, M., Nagata, K.: Quantum selective encryption for medical images. Int. J. Theor. Phys. 58(11), 3908–3926 (2019)
15. Henderson, M., Shakya, S., Pradhan, S., Cook, T.: Quanvolutional neural networks: powering image recognition with quantum circuits. Quant. Mach. Intell. 2(1), 1–9 (2020). https://doi.org/10.1007/s42484-020-00012-y
16. Hoover, A., Kouznetsova, V., Goldbaum, M.: Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response. IEEE Trans. Med. Imaging 19(3), 203–210 (2000)
17. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708 (2017)
18. Jiang, Z., Zhang, H., Wang, Y., Ko, S.B.: Retinal blood vessel segmentation using fully convolutional network with transfer learning. Comput. Med. Imaging Graph. 68, 1–15 (2018)
19. Jin, Q., Meng, Z., Pham, T.D., Chen, Q., Wei, L., Su, R.: DUNet: a deformable network for retinal vessel segmentation. Knowl.-Based Syst. 178, 149–162 (2019)
20. Khan, M.A.U., Khan, T.M., Bailey, D.G., Soomro, T.A.: A generalized multiscale line-detection method to boost retinal vessel segmentation sensitivity. Pattern Anal. Appl. 22(3), 1177–1196 (2018). https://doi.org/10.1007/s10044-018-0696-1
21. Li, L., Verma, M., Nakashima, Y., Nagahara, H., Kawasaki, R.: Iternet: retinal image segmentation utilizing structural redundancy in vessel networks. In: The IEEE Winter Conference on Applications of Computer Vision, pp. 3656–3665 (2020)
22. Liew, G., Wang, J.J.: Retinal vascular signs: a window to the heart? Revista Española de Cardiología 64(6), 515–521 (2011)
23. Liskowski, P., Krawiec, K.: Segmenting retinal blood vessels with deep neural networks. IEEE Trans. Med. Imaging 35(11), 2369–2380 (2016)
24. Maharana, D.K., Das, P.: Automatic extraction of vessels from newly accessible dataset. In: Soft Computing: Theories and Applications, pp. 1139–1150. Springer (2020). https://doi.org/10.1007/978-981-15-0751-9_105
25. Mo, J., Zhang, L.: Multi-level deep supervised networks for retinal vessel segmentation. Int. J. Comput. Assist. Radiol. Surg. 12(12), 2181–2193 (2017). https://doi.org/10.1007/s11548-017-1619-0
26. Pal, S., Chatterjee, S., Dey, D., Munshi, S.: Morphological operations with iterative rotation of structuring elements for segmentation of retinal vessel structures. Multidimens. Syst. Signal Process. 30(1), 373–389 (2018). https://doi.org/10.1007/s11045-018-0561-9
27. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015 Part III. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
28. Sazak, Ç., Nelson, C.J., Obara, B.: The multiscale bowler-hat transform for blood vessel enhancement in retinal images. Pattern Recognit. 88, 739–750 (2019)
29. Birgui Sekou, T., Hidane, M., Olivier, J., Cardot, H.: Retinal blood vessel segmentation using a fully convolutional network – transfer learning from patch- to image-level. In: Shi, Y., Suk, H.-I., Liu, M. (eds.) MLMI 2018. LNCS, vol. 11046, pp. 170–178. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00919-9_20
30. Simard, P.Y., Steinkraus, D., Platt, J.C., et al.: Best practices for convolutional neural networks applied to visual document analysis. In: ICDAR, vol. 3 (2003)
31. Smith, L.N.: Cyclical learning rates for training neural networks. In: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 464–472. IEEE (2017)
32. Soomro, T.A., Khan, T.M., Khan, M.A., Gao, J., Paul, M., Zheng, L.: Impact of ICA-based image enhancement technique on retinal blood vessels segmentation. IEEE Access 6, 3524–3538 (2018)
33. Staal, J., Abramoff, M., Niemeijer, M., Viergever, M., van Ginneken, B.: Ridge based vessel segmentation in color images of the retina. IEEE Trans. Med. Imaging 23(4), 501–509 (2004)
34. Tamim, N., Elshrkawey, M., Abdel Azim, G., Nassar, H.: Retinal blood vessel segmentation using hybrid features and multi-layer perceptron neural networks. Symmetry 12(6), 894 (2020)
35. Trujillo, M.C.R., Alarcón, T.E., Dalmau, O.S., Ojeda, A.Z.: Segmentation of carbon nanotube images through an artificial neural network. Soft Comput. 21(3), 611–625 (2017)
36. Wang, X., Jiang, X.: Retinal vessel segmentation by a divide-and-conquer funnel-structured classification framework. Signal Process. 165, 104–114 (2019)
37. Zapata, M.A., et al.: Artificial intelligence to identify retinal fundus images, quality validation, laterality evaluation, macular degeneration, and suspected glaucoma. Clin. Ophthalmol. 14, 419 (2020). (Auckland, NZ)
38. Zhang, A., Wang, K.C., Yang, E., Li, J.Q., Chen, C., Qiu, Y.: Pavement lane marking detection using matched filter. Measurement 130, 105–117 (2018)
39. Zhang, Z., Liu, Q., Wang, Y.: Road extraction by deep residual u-net. IEEE Geosci. Remote Sens. Lett. 15(5), 749–753 (2018)
40. ZhaoZhao, J., et al.: Automatic retinal vessel segmentation using multi-scale super-pixel chain tracking. Digit. Signal Process. 81, 26–42 (2018)
41. Zhou, M., Jin, K., Wang, S., Ye, J., Qian, D.: Color retinal image enhancement based on luminosity and contrast adjustment. IEEE Trans. Biomed. Eng. 65(3), 521–527 (2017)