

Fast and efficient retinal blood vessel segmentation method based on deep learning network

Henda Boudegga, Yaroub Elloumi, Mohamed Akil, Mohamed Hedi Bedoui, Rostom Kachouri, Asma Ben Abdallah

▶ **To cite this version:**

Henda Boudegga, Yaroub Elloumi, Mohamed Akil, Mohamed Hedi Bedoui, Rostom Kachouri, et al.. Fast and efficient retinal blood vessel segmentation method based on deep learning network. Computerized Medical Imaging and Graphics, 2021, 90, pp.101902. 10.1016/j.compmedimag.2021.101902 . hal-03348490

HAL Id: hal-03348490
https://hal.science/hal-03348490

Submitted on 27 Sep 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Fast and efficient retinal blood vessel segmentation method based on deep learning network

Henda Boudegga[a,c], Yaroub Elloumi[a,b,c], Mohamed Akil[b], Mohamed Hedi Bedoui[a], Rostom Kachouri[b], Asma Ben Abdallah[a]

[a]Medical Technology and Image Processing Laboratory, Faculty of medicine, University of Monastir, Tunisia.

[b]LIGM, Univ Gustave Eiffel, CNRS, ESIEE Paris, F-77454 Marne-la-Vallée France

[c] ISITCom Hammam-Sousse, University of Sousse, Tunisia.

ABSTRACT

The segmentation of the retinal vascular tree presents a major step for detecting ocular pathologies. The clinical context expects higher segmentation performance with a reduced processing time. For higher accurate segmentation, several automated methods have been based on Deep Learning (DL) networks. However, the used convolutional layers bring to a higher computational complexity and so for execution times. For such need, this work presents a new DL based method for retinal vessel tree segmentation. Our main contribution consists in suggesting a new U-form DL architecture using lightweight convolution blocks in order to preserve a higher segmentation performance while reducing the computational complexity. As a second main contribution, preprocessing and data augmentation steps are proposed with respect to the retinal image and blood vessel characteristics. The proposed method is tested on DRIVE and STARE databases, which can achieve a better trade-off between the retinal blood vessel detection rate and the detection time with average accuracy of 0.978 and 0.98 in 0.59s and 0.48s per fundus image on GPU NVIDIA GTX 980 platforms, respectively for DRIVE and STARE database fundus images.

1. Introduction

The retina is the eye sensory membrane which is responsible for vision. It contains the mainly anatomical components which are the Optic Disc (OD), the macula and the blood vessels, as shown in Fig.1. The retinal blood vessels converge to the center of the OD and spread into the whole retina, except the macula, with a thickness reduction. Several ocular pathologies alter the morphology of the vascular tree. The proliferative stage of diabetic retinopathy [1, 2], or the wet stage of Age-related Macular Generation (AMD) [3] lead to adding new vessels with a lower thickness, tortured form and lower contrast with respect to the predecessor ones. In the non-proliferative stage of diabetic retinopathy and hypertensive retinopathy [4], vessel blood leakage cause the appearance of numerous lesions close to vessel tree such as micro-aneurysms [5, 6]. For this purpose, the retinal vessel tree is considered as a principle biomarker of several ocular diseases where screening and severity grading have been always predicted by referring to the modification of vessel morphology.

The main challenge is to perform a precise and accurate segmentation of the retinal vascular tree, to be useful for the ophthalmological diagnostic. In this context, several automatic methods for retinal vessel segmentation have been put forward, which are reported in several reviews [7, 8, 9]. Those studies have proposed to classify methods based on their processing principles. The image processing based methods can be based on thresholding [10, 11, 12, 13, 14, 15, 16], filtering [17] [18] or vessel tracking techniques [23, 24].

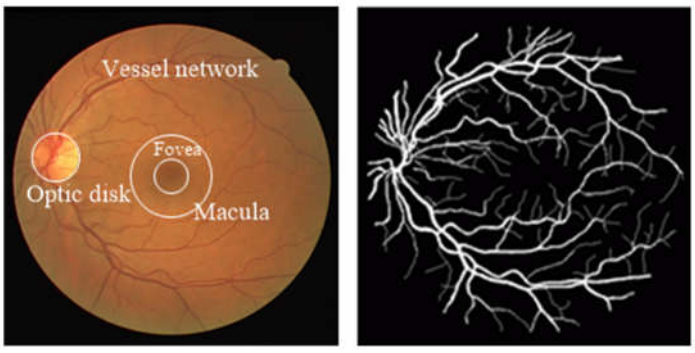


Fig. 1. Anatomy of retina: (left) retinal image; (right) segmented image.

Several recent methods have been based on Machine Learning (ML) which consists in training models in order to provide classification as vessel and non-vessel pixels [21, 22, 23, 24, 25]. Other methods based on Convolution Neural Networks (CNNs) architectures are introduced. Those methods consist either to applying well-defined networks such as AlexNet [26] and U-Net [27], or extending existing networks with the aim of enhancing segmentation performance such as [27, 28, 29, 30, 31]. Those segmentation vessel tree methods have achieved a higher segmentation performance where the accuracy rate is in the order of 0.91 and 0.98, the sensitivity rate in the order of 0.64 and 0.83 and the specificity rate in the order of 0.93 and 0.98.

Additionally, the speed of the retinal tree segmentation is an important criterion for the ophthalmological diagnostic, which is constrained by the large number of people to diagnosis, the permanent increase in the filed-of-view and in the resolution of fundus images, both caused by the imaging technologies progress. However, several methods have not addressed the computational criteria, such as the methods suggested in [32, 21] where their computation times were expressed in terms of minutes, as shown in Table 1. Despite their higher segmentation performance, a raised execution time is a limiting factor to implement those segmentation methods on a CAD system for ocular pathology detection. Numerous methods have aimed to optimize the processing complexity [33], [34], where the segmentation is reached into reduced execution times, as reported in the first two rows of Table 1. However, the fast segmentation has been associated to lower detection rates respectively in the range of 0.918 and 0.938. Otherwise, several methods have aimed at insuring efficient detection rates with lower time consumption such as the work of [17, 35, 36], where their computation times above one second, reached respectively in a quad cores, single core and on multi-GPU architectures.

Particularly, the DL-based methods have been always belonged to the studies providing higher segmentation performances [37, 27, 38]. Their architectures are based on convolution blocks that have ensured extracting vessel features even with the complicated details of retinal vascular tree into the retinal texture. They have provided higher accuracy rates that exceed 0.95 when performed with the well-used DRIVE database fundus images. Moreover, the permanent rise of fundus image resolution allows illustrating a large gap of vessels details. Hence, DL networks will be heavily needed with the aim of guaranteeing accurate results.

This work was supported by the PHC-UTIQUE 19G1408 Research program.

However, the expended convolution blocks correspond to a filter processing where convolution kernels are combined into one procedure. Each kernel is performed through a wide amount of arithmetic processing. Hence, those blocks are discriminated by a higher computational complexity which is proportional to the number of parameters. The networks used in [39, 40] have produced respectively 31 and 48 million parameters. Consequently, the rise on the complexity and on the parameter number leads a similar rise on their computation times. As the case of the networks expended in the studies [27, 40], the segmentation has been produced respectively in 15.3s and 96s, despite that have been both executed into recent multi-GPU architectures. with respect to the accurately and computational constraints, several segmentation methods suggest extending well-known networks such as U-net in the works [27, 28, 29], Alexnet [26], VGG [41] and Z-Fnet[42] [94]. However, these network extensions still unable to reach the targeted trade-off. They are always discriminated either by inaccurately segmentation results which is inappropriate for pathologies diagnosis, either by a high execution time which does not respect clinical requirements. In addition, these segmentation networks suffering from high computation time, caused by the higher computational complexity of the convolution layers exposed in their proposed networks.

In this work, our challenge is to put forward a new deep neural network that ensures an accurate segmentation of the retinal vessel tree into a reduced computation time. Within this objective, we propound a novel DL architecture by expending Lightweight Convolution Modules (LCMs) characterized by lower complexity with respect to standard convolution. In addition, a preprocessing is performed in order to improve the quality of images and enhance the contrast of vessels in the retina. Thereafter, data augmentation is proposed to transform and crop images, in order to guarantee training robustness. Next, these patches of images are resized to be adapted to our suggested DL architecture. The segmented patches are then merged to generate the segmented blood vessel image.

The remainder of this paper is organized into three sections. In section 2, the principle and processing pipeline of the proposed method are described. Section 3 presents the experimentation results where the segmentation is evaluated in terms of performance and execution time and compared to existing methods. The last section is dedicated for conclusion.

2. DL architecture for retinal vessel tree segmentation

2.1. *Principle of our proposed method*

In this work, an adequate tradeoff between segmentation performance and execution time is our important intention. An accurate segmentation of retinal vascular tree is indispensable for a reliable diagnosis of ocular pathologies. Additional, performing efficient segmentation while saving computation performances is indispensible to be suitable for clinical practice.

Table 1
Performances results of state of the art works

Works	Time(s)	Acc	plateform
Biblioni et al. [33]	0.037	0.938	Quad CPU cores 3.10 GHz, 16 GB RAM
Alonso-Montes et al. [34]	0.193	0.918	SIMD parallel processor
Arguello et al. [44]	0.421	0.943	Quad CPU cores 3.10 GHz, 16 GB RAM
Budai et al. [35]	1.04	0.957	One CPU cores 2.3GHz, 4GB RAM
Jiang et al. [17]	1.677	0.9597	Quad CPU cores 2.3GHz, 16 GB RAM
Zhu et al. [23]	12.16	0.9607	Quad CPU cores 4.0Ghz, 32GB RAM
Jin et al. [27]	15.3	0.9566	NVIDIA GeForce GTX 1080Ti GPU
Sudeshna et al. [32]	148.48	0.9628	One CPU cores 1.5 GHz, 512 MB RAM
Geethramani et al. [21]	1200	0.9536	Quad CPU cores, 2.53 GHz.
Liskowski et al. [40]	92	0.9515	NVIDIA GTX Titan GPU

Recently, several Convolution Neural Networks (CNN) have been proposed for the segmentation issues such as Alexnet [43], U-net [39], VGG [44], where various have been extended with the aim of performing such clinical needs. Those networks still unable to reach the trade-off performances, discriminated by an expensive convolution blocks having the effect of increasing computation requirements. In fact, the CNNs have been continuously made development, where several networks aimed to reducing computational complexity. Their main contributions consist at suggesting new convolution blocks that ensures the same convolution processing with reduced complexity.

Our main idea consists at proposing a new DL network which performs retinal vessel segmentation using LCMs, as explicitly detailed in section 2.4. The suggested network guarantees a higher segmentation performance of the retinal vascular tree within a decreased computation time, which allows reaching the aimed tradeoff. Accordingly, our method for retinal vessel segmentation is based on the trained model of the proposed network, where its processing pipeline is put forward as presented in the diagram of Fig.2.

The retinal blood vessels are presented in the retina as thin, elongated and dark structures [24, 34] which spread into the retina with thickness variation [45]. In contrast, the fundus image is characterized by an imbalanced contrast between blood vessels and retinal background, where the color of thin vessels may be close to the color of the background [45, 46]. For this, the fundus images will be preprocessed in order to improve the quality of images and enhance the contrast of vessels, as described in section 2.2. Further, the retinal blood vessels have a wide variety of configurations in terms of thickness, direction, tortuosity, etc. The vessel shape varieties are represented inadequately, where thick vessels are always illustrated in horizontal direction. Furthermore, they have

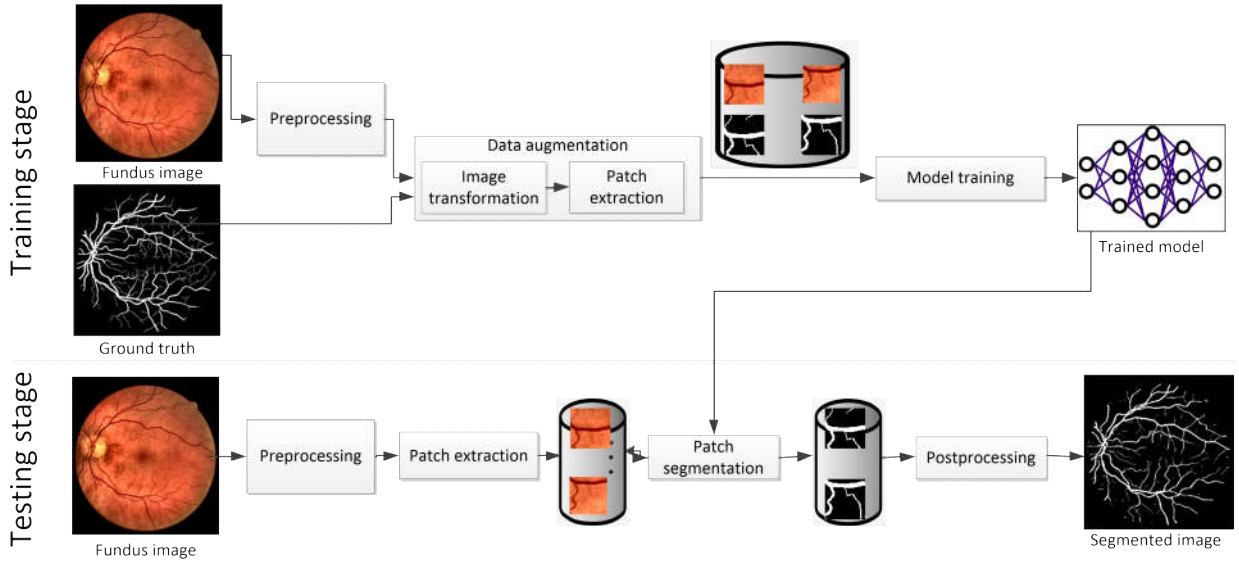


Fig. 2. Workflow of proposed retinal blood vessel segmentation method.

vertical direction in the OD where bifurcations are well distinguished. Therefore, a data augmentation process is developed throw applying fundus images transformation in order to generate additional examples with the aim of improving method robustness, as depicted in section 2.3.1. It should be noted that inaccurate results might be provided when segmenting an entire image through the DL network. Thus, a cropping step is required to crop images into patches, to be used for training the suggested neural network, as described in 2.3.2. This step will be reproduced in testing stage to generate patches that are segmented through the trained model. Subsequently, the segmented patches are merged through the post processing step to provide the segmented vessel tree, as illustrated in the testing stage of Fig.2.

2.2. Preprocessing

This step aims to improve the fundus image quality and differentiate vessels from the background in order to achieve accurate segmentation. In fact, the retina is encircled by a black area. Since the background intensity is close to the blood vessel one, it leads to distort the segmentation. Therefore, we firstly replace the black area by the average color of the retina [17, 26], where the result is shown in Fig.3 (b). Subsequently, the "L" channel provides better contrast compared to the A and B channels, where the L channel represents the luminosity layer and the A and B channels represent the chromaticity ones [47]. Therefore, the RGB image is converted to a LAB color space. After that, Contrast Limited Adaptive Histogram Equalization (CLAHE) is applied on the L channel [47, 48], in order to enhance contrast. Thereafter, the adjusted L channel is merged with the A and B channels, as depicted in Fig.3 (e), and then reconverted to an RGB format with the superposition of a mask [26], where the final result is given in Fig.3 (f).

2.3. Data augmentation

2.3.1. *Image transformation*

The configuration of the retinal vascular tree is too varied in terms of thickness, direction, tortuosity, etc. In this context, a data augmentation process is proposed, which consists in providing fundus images by modifying the existing ones. This step leads to raise the data set size to increase the robustness of the suggested DL model. As regards vessel orientation and thickness, each fundus image is rotated four times, where the rotation angles are 30◦, 60◦, 120◦ and 150◦, as presented in Fig.4 (e, e')-(h, h'). Furthermore, the fundus image is flipped respectively in horizontal, vertical, and both horizontal and vertical directions. These transformations are performed for all fundus images in the same dataset, either for preprocessed input images or segmented output ones to avoid distorting training, as provided in Fig.4 (a, a')-(d, d).

2.3.2. *Cropping*

Retinal fundus images are characterized by an important number of blood vessels which are represented with different thicknesses, orientations and tortuosities. As a result, the segmentation of an entire image through a DL network remains a hard task, where inaccurate results might be generated. Therefore, a squaring process is proposed as in [40, 37, 26], which consists in cropping the fundus image into patches containing a partial blood vessel tree, to be segmented separately, as shown in Fig.5. The patch size "μ" is identified to enhance the segmentation performance in a reduced computation time. In fact, fundus image databases are diverged in terms of size and resolution. Hence, the patch size "μ" must be selected for each dataset separately, which are identified experimentally. For that purpose, the fundus images of the same database are cropped into several patches with a reduced "μ" size to be used for training. This step is performed iteratively while increasing the cropping size and keeping the other training parameters. Thereafter, we choose the "μ" value leading to the highest segmentation performance.

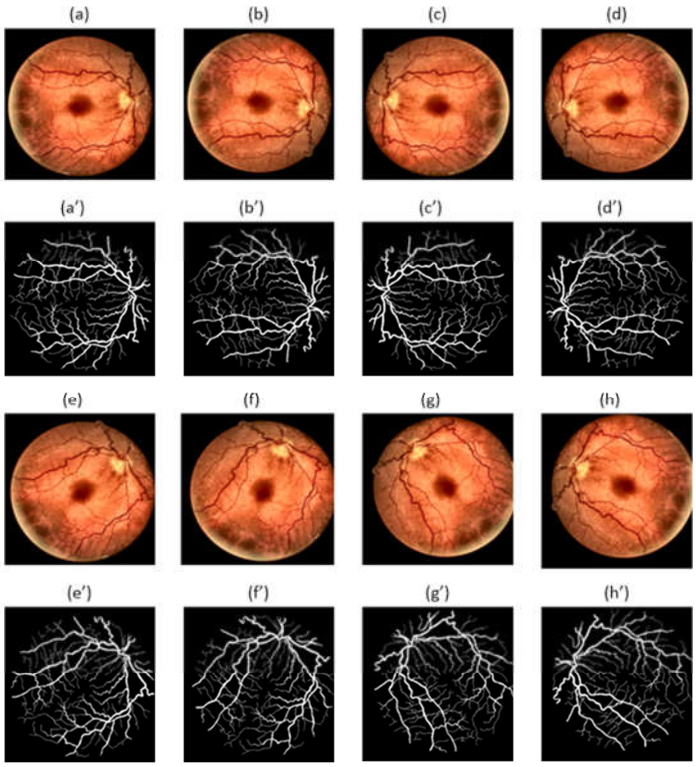


Fig. 4. Retinal image transformation.

2.4. RV-Net: Proposed network for retinal blood vessel segmentation

2.4.1. *Network description*

With the aim of guaranteeing accurate segmentation results and improving the inference efficiency, we propose a new architecture untitled "RV-Net" for retinal vessel tree segmentation. The proposed network should have a U-form with downsampling and upsamling paths. This form is widely adopted for the segmentation issue and reached higher performances. Further, this form encourages the model to learn more features through gradually reducing feature maps and making connection of the both sides. Within this context, we suggest extending the well-known network U-net, that ensured a higher performance segmentation in several biomedical issues [49, 50, 38]. Their blocks are mainly based on standard convolution layers. Each layer consists at performing filtering and combining convolutional kernels into one step, which leads to an augmented processing complexity and so for the whole architecture. Recently, LCMs have been propounded which ensure

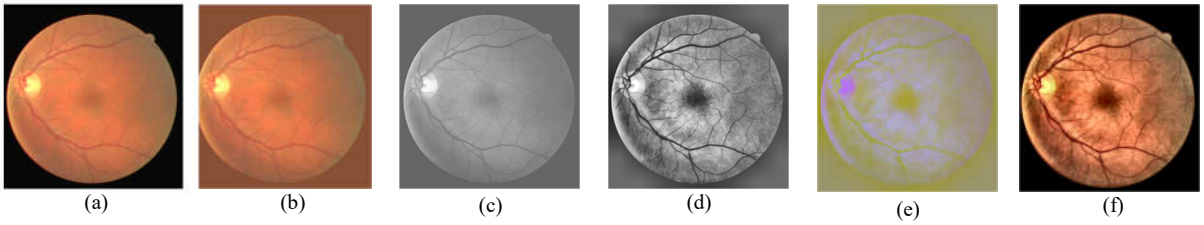


Fig. 3. Preprocessing steps: (a) original fundus image; (b) Black ring removed; (c) L channel; (d) L channel after CLAHE processing; (e) merged fundus image; (f) preprocessed fundus image.

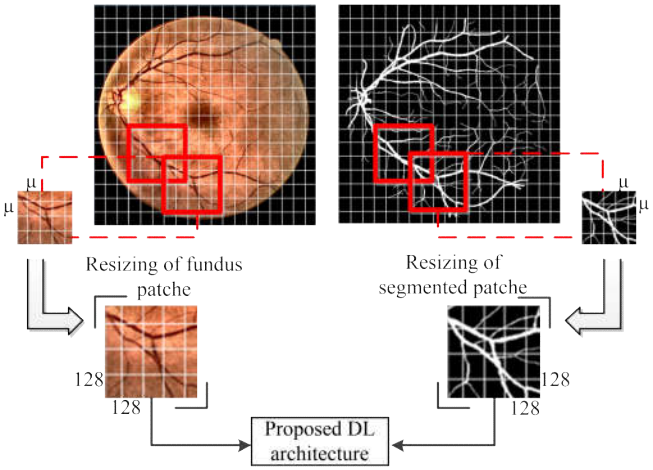


Fig. 5. Fundus image cropping and patch resizing.

the same convolution through split processing into two successive operations. Such modification reduces significantly the computation requirements, where the computation times might be reduced nine times with respect to standard convolution layers [51].

Hence, our main contribution consists of extending the well-known network U-net through consuming LCMs proposed by [51] instead of standard convolution layers, with the aim of guaranteeing accurate performance by adopting the U-form and reduce computing by using LCM. The used LCM consists of a "3x3 depthwise convolution layer" followed by the "1x1 convolution layer". The 3x3 depthwise convolution layer takes as an input M feature maps having the size of $W \times H$, where W and H represent respectively the width and the height of feature maps. Each feature map is extracted separately in order to apply iteratively a $K \times K \times 1$ convolution kernel, where K represents the kernel width and height. Hence, M feature maps are provided through the first layer, having the dimension of $W \times H \times 1$. For the second layer, the M feature maps are superimposed, on which a $1 \times 1 \times N$ kernel is iteratively applied to provide N feature maps.

The downsampling path of the suggested network is composed by six blocks, which are represented with red frames in Fig.6. The first block is composed by a convolution layer using a 3x3 kernel to produce 64 feature maps. Then, a LCM is applied where his first layer has convolved 64 feature maps separately, using 3x3 kernels. Then, a convolution layer with 1x1 kernels, are applied to produce 128 feature maps. Those three convolution layers are characterized by a stride equal to 1 and an activation function RELU. This module is followed by a "max pooling" layer parameterized with a 2x2 kernel and a stride of 2, as shown in red in Fig.6.

For the other downsampling blocks, each one is composed by two LCMs followed by a " max pooling" layer parametrized by a $\mathbf{2 \times 2}$ kernel and a stride equal to 2 as illustrated in Fig.6, except the last block which contains only two LCMs. The number of output feature maps is doubled and their output sizes are reduced by half on the length, as well as on the width, providing N feature maps having the size of $W/2 \times H/2$, as noted above each blocks in Fig.6.

For the case of the second block of the downsampling path, M feature maps matching 128 channels with W=H=64 have been convolved through the first LCM to produce 128 feature maps having the same dimensions, as shown in Fig.7. The output is fed to the second LCM in order to generate N feature maps matching 256 channels with W=H=64. Thereafter, the produced output is provided to a max-pooling processing generating the same input feature maps number equal to 256 having the size of $64/2 \times 64/2$.

Similarly, the upsampling path is composed by five blocks. Each block contains an upsampling layer parameterized with a 2x2 kernel and a stride of 2, as shown with green in Fig.6. For each block, the upsampling layers are followed by two LCMs. Their first depthwise convolution layers are parameterized with a 3x3 kernel, where their second convolution layers are parameterized with 1x1 kernels. These convolution layers are parameterized with an activation function RELU and a stride equal to 1. Contrary to the downsampling path, the number of output feature map is reduced in each block by half and the output size of the feature map is doubled on the length, as well as on the width. The network is concluded by adding a convolution layer with two kernels of size 1×1 and a standard softmax activation layer, which performs 64 feature maps of the vessel and background classes. Furthermore, each block of the upsampling path is concatenated with its corresponding of the downsampling path which has the same size and the same feature map numbers.

2.4.2. *Parameter setting*

The training of the proposed network is done using a set of parameters chosen experimentally or by referring to recent studies with the aim of providing accurate segmentation performance. The learning rate, the optimizer algorithm, the weight initialization technique and the epoch number are determined experimentally. For each parameter, we perform some training experience for a single model while modifying only the parameter value. Thereafter, we choose the value offering the higher segmentation rates. As the case of the learning rate and the optimizer algorithm, the experimentation consists at varying the parameter value and evaluated the results as shown in Fig.8(a) and Fig.8(b), where the suitable parameter providing the higher performances are respectively the ADAM optimizer with 0.0001 value of learning rate. Furthermore, we propound initializing the weight using Xavier technique and minimizing the loss between predicted images and the ground truth using cross entropy function and setting the dropout ratio to 0.5 for regularizing training experiments. All the selected parameters are summarized in Table 2.

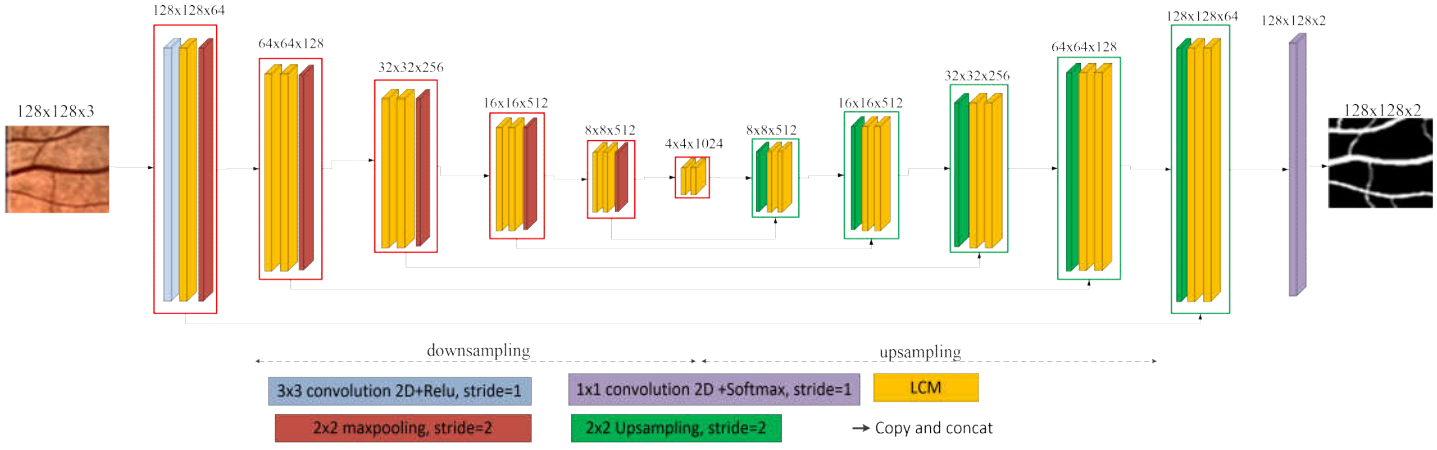


Fig. 6. RV-Net proposed DL network for retinal blood vessel segmentation.

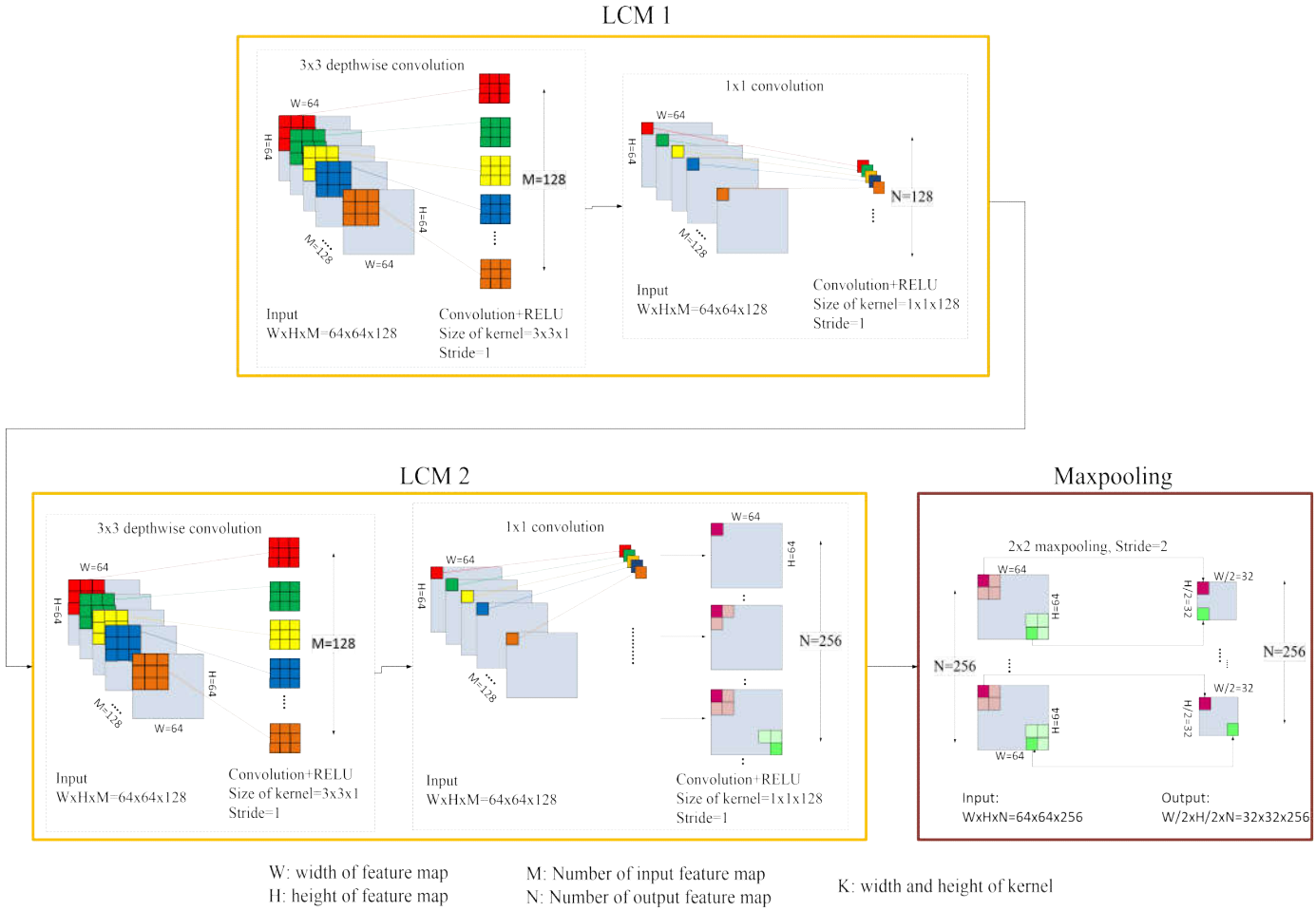
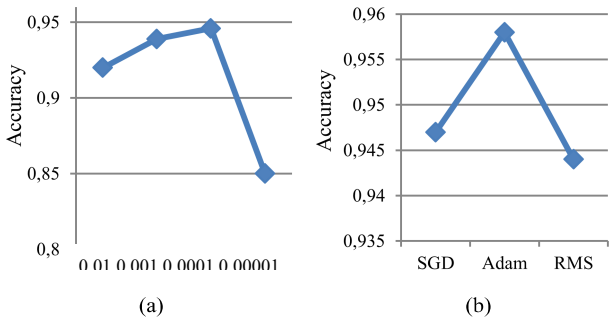


Fig. 7. Structure of the second Downsampling block of the proposed network: LCM 1 and LCM 2 represent respectively the both modules of the second Downsampling block.



(a) (b)

Fig.8. Performance measurement with different values of learning rate and optimization algorithm.

.

Table2.
Training parameters

Parameter	Value
Optimizer	Adam optimizer
Learning rate	0.0001
initialization technique	Xavier
Regularization	dropout technique(0.5)
Epoch number	40
Loss function	cross entropy

2.4.3. *Computation complexity of the pro posed network:*

We propose in this section to study the complexity of the proposed network, with the aim of evaluating the computation efficiency throw using the LCMs instead to convolution layers. For a standard convolution layer, a single filter having a size of $K \times K \times M$ is convolved to an input map-extracted square having the same size, where K present the width and the height of the used kernel and M the number of feature maps. The aim is to provide a single value in the output feature map. Hence, the filter computational complexity is about $K \times K \times M$. The filter application is performed iteratively $W \times H \times N$ times to provide all the values of the output feature map, where W and H corresponds respectively to the width and the height of the feature maps and N the number of output feature maps. Therefore, the computation complexity of one convolution layer is indicated in Equation (1).

In contrast, the LCM is convolved into two steps. For the depthwise convolution layer, each feature maps is extracted separately, as outlined in Fig.7. For each feature maps, a single filter having size $K \times K \times 1$ is convolved to an input-extracted square having the same size as the used filter in order to provide a single value. The filter processing is performed iteratively $W \times H \times 1$ times in respect of each square of a single feature maps of the input feature map. Thus the computation complexity in the order of $K \times K \times W \times H$. Taking into account the M feature maps, the computation complexity of the depthwise convolution layer is about $M \times K \times K \times W \times H$. For the second layer of the LCM illustrated in Fig.7, the M feature maps are superimposed, on which a filter having size $1 \times 1 \times N$ is applied to provide a single value of the output feature map. This filter is applied iteratively $W \times H \times N$ times as regards the output feature map size. As a consequence, the computation complexity of the pointwise convolution layer is in the order of $N \times M \times W \times H$. Accordingly, the computation cost of the LCM is defined in Equation (2).

$$C_{\text{convolution layer}} = O(M \times N \times W \times H \times K^2) \quad (1)$$

$$C_{LCM} = O(M \times H \times W \times K^2 + M \times N \times W \times H) \quad (2)$$

Consequently, the LCM is considerably more computationally efficient than the convolution layer where complexity is decreased by $O(\max(N, K^2))$. As the proposed network is composed by L blocks and each block contains 2 LCMs, the utilization of LCMs instead of convolution layers allows reaching a gain of computational complexity in the order of $O(2 \times L \times (\max(N, K^2)))$.

2.5. *Post processing*

The goal of post processing is to merge all the segmented patches in order to generate a single segmented blood vessel image, as depicted in Fig.9. Therefore, the provided patches are collected and resized to their cropping size. Afterwards, these patches are copied in the order to perform the cropping, with respect to the image size. Subsequently, the mask of the used image is superimposed on the merged one in order to eliminate the white pixels around the retina. Then, the morphological transformation "erosion" is applied to eliminate noise by using an ellipse structuring element with size 2×2.

3. Experiments and analysis

The method is configured and implemented where the training parameters and the execution platform are described in section 3.1. To evaluate the proposed method with respect to its background, the experiment is performed using well-known retinal image databases to provide the evaluation metrics described in section 3.2. The experiment is carried out by distinctly evaluating the segmentation performance, the computation requirements and the tradeoff. These evaluation performances are presented respectively in section 3.3, section 3.4 and section 3.5, in which a comparative assessment with respect to the state-of-the-art methods is suggested.

3.1. *Experiment setup*

The validation of the proposed method is performed using the two publicly retinal image databases DRIVE and STARE. Both databases contain respectively 40 and 20 images having sizes of 565×584 and 700×605. These databases contain respectively seven and ten pathological fundus images affected by the diabetic retinopathy [8]. For these databases, each retinal image is joined with its manual blood vessel segmentation. The Field Of View (FOV) mask is provided for the DRIVE database, while the one corresponding to the STARE database is created by applying a simple threshold to the gray channel. The implementation is conducted on Intel core i7 with a 3.67 GHZ frequency processor, 8Go RAM and a NVIDIA GTX 980 GPU. The suggested method is implemented using Python 3.5.2 as a programing language, with OpenCV library 3.4 and Tensorflow framework 1.12. Tensorflow and OpenCV are open source libraries used respectively for image processing and computer vision applications [52] and for expressing machine learning algorithms [53]. The GPU execution is performed using CUDA 9.0 with CUDNN 7.6.3.

3.2. *Evaluation principle and metrics*

In order to efficiently evaluate the suggested method, we put forward a 4-fold cross validation approach, which consists in partitioning the retinal images of each database into four subsets in order to perform four experiments for each database. The goal of these experiments is to evaluate the robustness of the suggested method. For each experiment, three subsets are conducted for the training process and one subset for testing. Fig.10 shows the dispatching of subsets between training and testing of each experiment. For each experiment, a DL segmentation model is generated, where examples of segmentation results of DRIVE and STARE retinal images are shown in Fig.11.

Additional, we suggest evaluating the segmentation results with respect to manual segmentation performed by an expert clinician. The evaluation is based on the classification of each pixel into True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). The TP and TN are respectively the number of pixels which are correctly classified as vessels and background, whereas the FP and FN are respectively the number of pixels misclassified as vessels and background.

The segmentation performance measures are summarized in Accuracy (Acc), Sensitivity (Se), Specificity (Sp), DICE coefficient, Precision. These metrics are the most frequently adopted measures for the evaluation of segmentation results. The ACC represents the ability to correctly classify pixels as background and vessels, while the SE and SP reflect respectively the capacity to correctly classifying pixels as vessels and the capability of correctly classifying pixels as background. The DICE coefficient represents the similarity between the ground truth and the segmented image, where the Precision indicates the proportion of correctly classifying pixels as vessels among all the correctly classified pixels as vessels as well as background. Further, we propound computing the statistical test P-Values with a confidence level of 95%. This test is used to determine whether there is a related significant performance between the 4 fold cross validation of two databases [54]. Table 3 summaries the performance metrics used by the proposed method.

3.3. *Segmentation performances*

In this section, we evaluate the segmentation performance of the proposed method using the 4-fold cross validation approach for

Table3
Performance metrics for the evaluation of retinal vessel segmentation.

Metrics	Description
Sensitivity (SE)	TP / (TP+FN)
Specificity (SP)	TN / (TN+FP)
Accuracy (ACC)	TN+TP/TP+FP+TN+FN
DICE	2*TP/ 2 * TP + F N + FP
Precision	TP/(TP+FP)

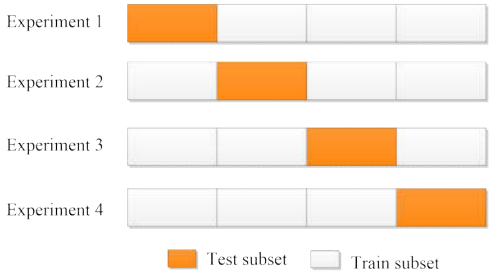


Fig. 10. Dispatching of subsets for 4-fold cross validation.

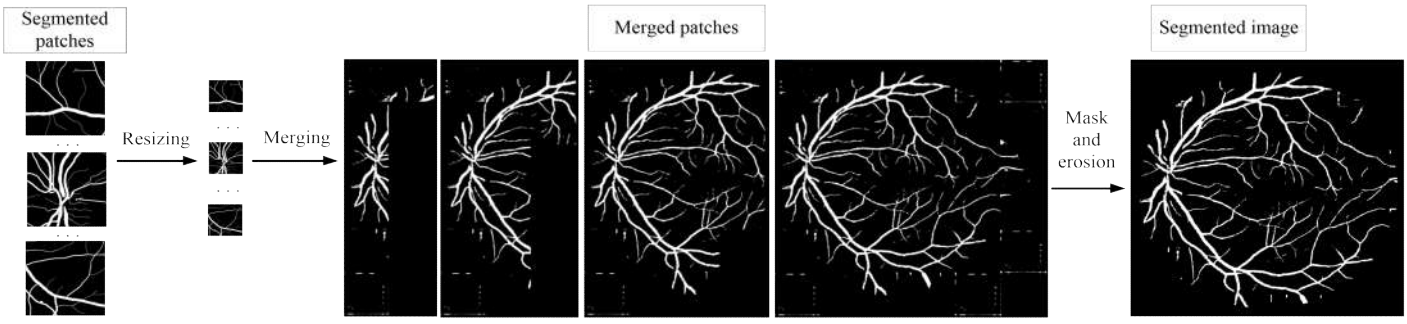


Fig. 9. Postprocessing.

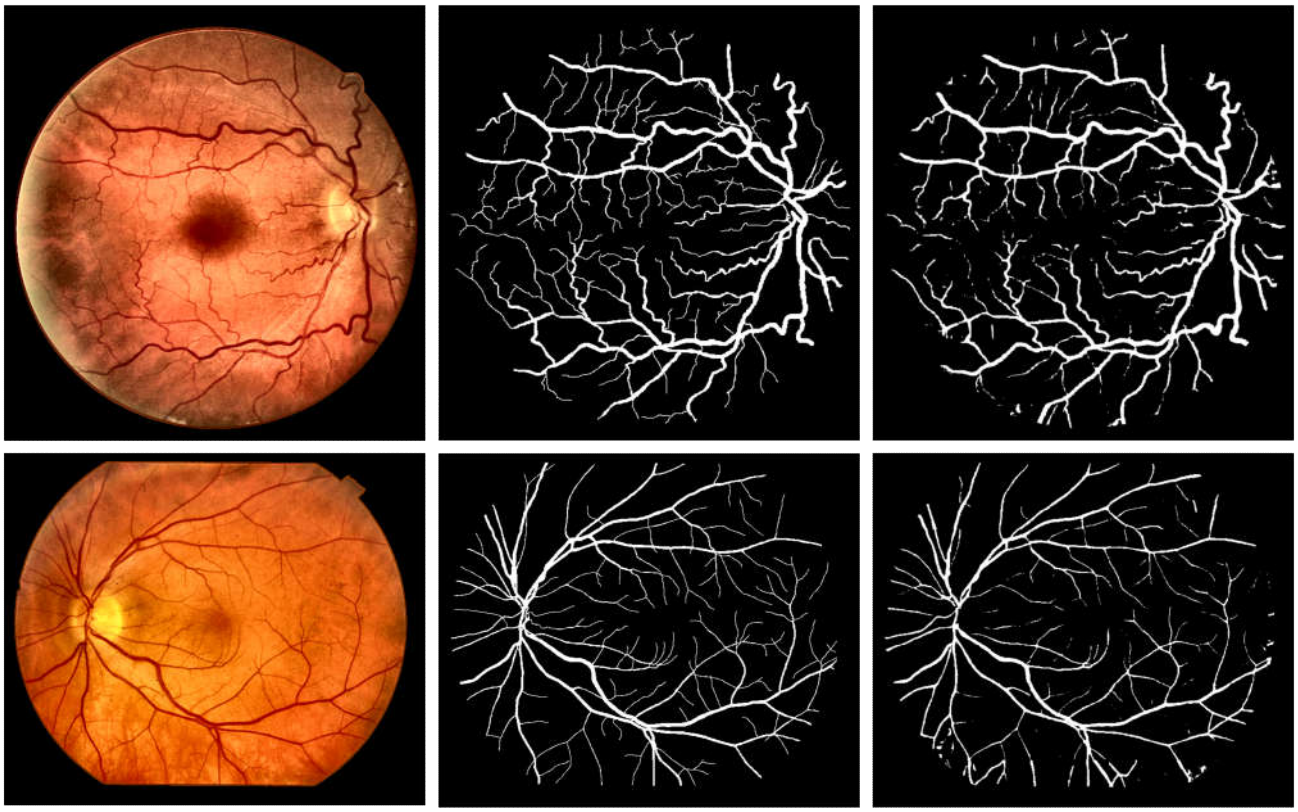


Fig. 11. Segmentation results: First row: DRIVE retinal image; Second row: STARE retinal image; First column: Preprocessed retinal images; Second column: Ground truth; Third column: Segmentation result.

each database based on ACC, SE, SP, DICE and Precision. The performance results are computed by referring to the manual segmentation provided by the expert. The performances measure of the 4-fold cross validation experiments are provided in Table 4. These experiments achieve a higher performance, where the average ACC, SE, SP, DICE and Precision are respectively in the order of 0.978, 0.8125, 0.9901, 0.8363 and 0.8680 for the DRIVE database and in the order of 0.98, 0.7913, 0.9905, 0.8156 and 0.8462 for the STARE database.

Moreover, the accuracy values provided by the 4-fold cross validation are very close to their average where the variation is in the order of 0.0024 and 0.0015 respectively for the DRIVE and STARE databases. Reduced gaps are also deduced between values for SE, SP, DICE and precision with variation values respectively in the order of 0.029, 0.0022, 0.0253 and 0.037 for the DRIVE database and in the order of 0.042, 0.023, 0.0259 and 0.02 for the STARE database. Consequently, the proposed method confirms a higher segmentation performance whatever the image used for the training or testing procedure. Additional, we suggest evaluating the correlation between the 4 fold-cross validation results of the two databases with the aim of testing the method robustness with respect to the used images. Thus, we propose computing the statistical tests P-values with the hypothesis that the results of the 4-fold cross validation are correlated. Therefore, we have applied the statistical inference test "Analysis Of Variance (ANOVA)" [55] basing on the accuracy values of the 4 fold cross validation. The Test return a P-values in the range of 0.077 for DRIVE and 0.36 for STARE, greater than the significant level (α=0.05). Hence, interpreting that 4-fold cross validation has related distribution.

Table 4
Average performance measure for 4-fold cross validation for both DRIVE and STARE database.

Database	Experiments	Metrics				
		Accuracy	Sensitivity	Specificity	DICE	Precision
DRIVE	1	0.98	0.8586	0.989	0.8583	0.859
	2	0.975	0.815	0.987	0.813	0.8173
	3	0.9819	0.8448	0.99	0.865	0.889
	4	0.977	0.7318	0.9946	0.8091	0.907
STARE	1	0.9784	0.7072	0.99	0.7638	0.83227
	2	0.9787	0.8523	0.987	0.8353	0.82
	3	0.9816	0.806	0.9928	0.839	0.8764
	4	0.9815	0.80	0.9924	0.8243	0.8563

Subsequently to evaluate the suggested method, we propose to compare our segmentation performance with the state-of-the-art methods for both DRIVE and STARE databases, where the ACC, SE and SP values are presented in Table 5. We deduce that our method achieves better accuracy than numerous methods such as [38] [26] [24] [42], [28] [56] and related sensitivity and specificity rates with respect to several works for both databases DRIVE and STARE databases.

3.4. Execution time performance

In this section, we examine the processing execution time of the suggested method. Accordingly, we propose to compute the execution time of each processing step for a single fundus image respectively from the DRIVE and STARE databases, as given in Table 6. We conclude that the computation times for the preprocessing, cropping and post-processing steps are too low despite the size of the used image. Besides, the segmentation step is performed in a short time, which is in the order of 0.6s and 0.48s respectively for the DRIVE and STARE images. This processing time is computed for all patches, where the elapsed times to segment a single patch is in the order of 0.0017±0.0003. To confirm the computation time behavior, we choose randomly 10 images respectively from both databases, in order to compute their whole execution times for vessel tree segmentation, as depicted in Fig.12. We deduce that the execution time is varied between 0.53 and 0.65 for the DRIVE images and between 0.42 and 0.57 for the STARE images, where the average computation times are respectively 0.59s and 0.5s, as shown in Fig.12 with red lines. Consequently, the computation times are considered as stationary whatever the used image is. This shorter computation time is explained by the suitable choice of the size of patches used for each size of image, which allows generating a limited number of patches, hence keeping the computation burden low.

Thereafter, we propose to evaluate the execution time in term of accuracy, in relation to the existing methods. The evaluation is performed with methods providing timing data. Both metrics

Table 5
Comparison of segmentation performances for DRIVE and STARE databases.

Works	Year	DRIVE			STARE		
		ACC	SE	SP	ACC	SE	SP
Condurache et al. [55]	2012	0.9516	**0.9094**	0.9591	0.9595	**0.8902**	0.9673
Roychowdhyry et al.[24]	2015	0.9519	0.7249	0.983	0.9515	0.7719	0.9726
Roychowdhury et al. [11]	2015	0.949	0.739	0.978	0.956	0.732	0.984
Imani et al.[18]	2015	0.9523	0.7524	0.9753	0.9590	0.7502	0.9745
Jiang et al.[17]	2017	0.9597	0.8375	0.9694	0.9579	0.7767	0.9705
Biswal et al.[19]	2018	0.95	0.71	0.97	0.95	0.70	0.97
Arguelo et al.[56]	2018	0.943	0.721	0.976	0.959	0.730	0.978
Sumathi et al.[57]	2018	0.9606	0.8014	0.9753	0.9435	**0.8339**	0.9536
Jiang et al.[26]	2018	0.9624	0.7540	0.9825	0.9734	**0.8352**	0.9846
Yan et al. [58]	2018	0.9538	0.7631	0.9820	0.9638	0.7735	0.9857
Wang et al.[28]	2019	0.9483	0.7886	0.9736	0.9475	0.7896	0.9734
Tang et al. [59]	2019	0.9574	**0.8564**	0.9710	0.9695	**0.8162**	0.9869
Jin et al.[27] U-Net	2019	0.9554	0.7849	0.9804	0.9637	0.76	0.9867
Jin et al.[27] D-U-Net	2019	0.9566	0.7963	0.98	0.9641	0.7595	0.9878
Tian et al.[38]	2020	0.958	**0.8639**	0.9690	-	-	-
Wu et al. [60]	2020	0.9582	0.7996	0.9813	0.9672	0.7963	0.9863
Nasser et al. [61]	2020	0.9607	0.7542	0.9843	0.9632	0.7806	0.9825
Proposed method	**2020**	**0.9819**	**0.8448**	**0.99**	**0.9816**	**0.806**	**0.9928**

Table 6
Running time partition for DRIVE and STARE images.

Processing steps	DRIVE	STARE
	Time(s)	Time(s)
Preprocessing	0.004	0.005
Cropping	0.005	0.012
Segmentation	0.6	0.48
	$(NP \times Time_per_patche)$	$(NP \times Time_per_patche)$
Postprocessing	0.00159	0.0018
Time per image(s)	0.61059	0.4988

NP: Number of Patches

correspond to the DRIVE database since it is the most used, where values are provided in Table 7. For that purpose, we investigate to split segmentation methods into two groups based on their execution time. These two groups correspond respectively to methods reaching an execution time below and above 1 second, as separated in Table 6 with bold line.

We deduce that first group put forward segmentation methods suggested by [33, 34, 57] which aim insuring fast computation time respectively in the range of 0.037s, 0.193s and 0.421s. In particular, our suggested method takes part of this group by providing segmentation results in 0.59s. In contrast, second group put forward studies suggested by [21, 23, 32] which suffer from higher computation times respectively in the range of 12.16, 1200 and 167 seconds. Additionally, while the suggested studies [33, 34, 57] reach the fast execution time, however, their accuracy rates were under 0.943. Meanwhile, our method is executed in 0.59s, while significantly achieving better accuracy in the order of 0.9819.

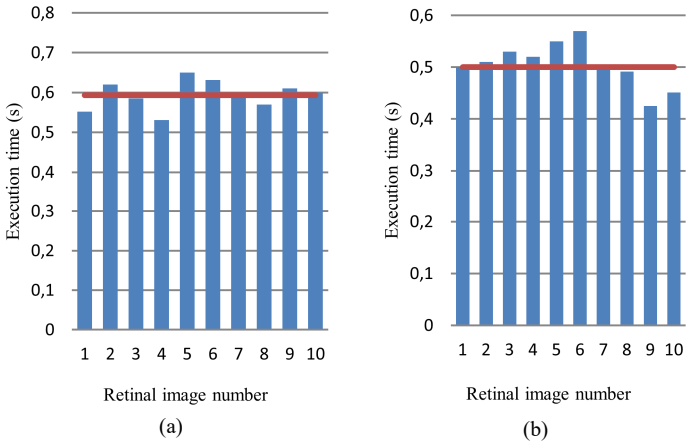


Fig. 12. Whole execution time for 10 images for each database: (a) DRIVE database; (b) STARE database.

3.5. tradeoff evaluation

In this section, we suggest firstly to study the tradeoff between computation requirement and segmentation results of U-Net based methods. We suggest investigating about the evolution of segmentation results according to computation requirement which consists of trainable parameters and execution time.

Table 7
Comparison of execution time and accuracy for DRIVE database.

Works	Year	Time(s)	ACC	Plateform
Alonso-Montes et al.[34]	2008	0.193	0.918	SIMD parallel processor
Arguello [57]	2018	0.421	0.943	Quad CPU cores 3.10 GHz, 16 GB RAM
Biblioni et al.[33]	2019	0.037	0.938	Quad CPU cores 3.10 GHz, 16 GB RAM
Proposed method	**2020**	**0.59**	**0.9819**	**Quad CPU cores 3.67 GHZ, 8GB RAM, GPU GeForce GTX 980**
Condurache et al. [56]	2012	6.8	0.9516	Dual CPU cores 2.66GHz,2GBRAM
Budai et.al [35]	2013	1.04	0.957	One CPU cores 2.3GHz, 4GB RAM
Krause et al.[36]	2013	1.2	0.9468	NVIDIA Geforce GTX680
Nuguyen et al. [58]	2013	2.5	0.94	Dual CPU cores 2.4 GHz, 2 GB RAM
Roychowdhury et al.[24]	2015	3.115	0.9519	Dual CPU cores, 2.6GHz and 2GB RAM
Roychowdhury et al.[11]	2015	2.45	0.949	Dual CPU cores 2.6GHz and 2GB RAM
Sudeshna et al.[32]	2016	148.48	0.9628	One CPU cores 1.5 GHz, 512 MB RAM
Geethramani et al. [21]	2016	1200	0.9536	Quad CPU cores 2.53 GHz 4 GB RAM
Jiang et al. [17]	2017	1.677	0.9597	Quad CPU cores 2.3GHz, 16 GB RAM
Zhu et al.[23]	2017	12.16	0.9607	Quad CPU cores 4.0Ghz, 32GB RAM
Alom et al.[29]	2018	6.4	0.9556	NIVIDIA GEFORCE GTX-980 Ti, 56G RAM,
Tang et al. [58]	2019	5	0.9574	Quad CPU cores 3.60 GHz, GPU GeForce GTX 1080TI
Jin et al.[27] U-net	2019	3.1	0.955	GeForce GTX 1080Ti GPU.
Jin et al.[27]D-U-net	2019	15.3	0.9566	GeForce GTX 1080Ti GPU.
Wu et al. [59]	2020	80	0.9582	E5-2690 V3, 2.6GHz, 128 GB RAM, GPU GeForce RTX 2080Ti

For this purpose, these computation requirements and segmentation results of U-Net based methods are identified, as presented in Table 8. Comparing to the U-Net network [39], our proposed model incurs a less parameter cost due to the LCMs used in the downsampling and upsamling blocks. Compared to U-Net extension based methods [42, 31], their DL models require lower parameter numbers with respect to ours, which are designed with only 128 filters and with specific convolution layers such as deformable layers and a residual layer. However, they are discriminated by an important execution time. Even so, our model is much faster reaching a computation time under 1 second, while performing better accuracy rate. Accordingly, our method offer the best tradeoff among the other methods through reaching lower parameter costs with respect to U-net, shorter execution time among those U-Net based methods and higher accuracy rate.

Additionally, we suggest studying the trade-off between the segmentation performance and the execution time of all retinal segmentation methods. For that purpose, we propose to investigate about the evolution of accuracy according to the execution time. For that, the retinal vessel segmentation methods are depicted into a 2D Cartesian space where horizontal and vertical coordinates correspond to the execution time and the accuracy rate, respectively.

We deduce that the studies [17, 29 29] aim enhancing segmentation accuracy while reducing the execution time have followed the same stability trend, as illustrated by the green slop in Fig.13. Particularly, our method reaches the shorter computation time with the highest accuracy rate among those methods, as represented by the blue point. For other work, the growth of the accuracy rate has been always associated with the rise in the execution time. Those methods can be partitioned into two trends, which are clustered around the red and orange slopes in Fig.13. Accordingly, an accuracy rate of 97% cannot be reached in less than 6s and 9s, respectively for the case of the orange and red slopes.

4. Conclusion

The segmentation of the retinal vessel tree is an indispensable step for the detection and diagnosis of various ocular pathologies such as diabetic retinopathy, age-related macular degeneration and glaucoma. This clinical context expects a higher segmentation performance on a reduced processing time. Within this objective, we have put forward a novel deep neural network architecture for the efficient segmentation of retinal blood vessels into reduced computation time. The suggested network consists of an extension of well-known network through expanding LCMs.

The proposed method is evaluated on both databases, DRIVE and STARE, with a 4-fold cross validation method. Thus, the suggested method has achieved high accuracy equal to 0.9819 and 0.9816 in 0.59s and 0.48s for the DRIVE and STARE databases, respectively. Our method has been evaluated with respect to the existing work, where it has ensured a better trade-off between accuracy and the execution time.

Otherwise, several automatic ophthalmological diagnostic systems such as [2, 59] are based on the segmentation of retinal vascular tree. However, they are discriminated by a higher execution time respectively in the range of 8.74s and 134.897s, where the computation time of vessel segmentation process are respectively in the order of 4.99s and 82.12s. In this context, our automated method can be directly employed to take benefit from its segmentation performance and computational time which are respectively in the range of 98% and 0.59s.

Besides, the suggested method can be extended for segmenting separately thick and thin vessel [24, 38]. Moreover, it can be useful for the identification of bifurcation points [60, 61] tortuosity [62] and thickness [63] measurement. The purpose is to ensure a better trade-off between accuracy and the execution time by providing higher detection performance into a shorter elapsed time.

Additionally, the provided results allow our dedicated future work to implement retinal vessel tree segmentation into mobile devices, as a higher interest in mobile health within the clinical context. Indeed, several mobile digital imaging devices have recently appeared, hence enabling the capture of fundus images with good quality on mobile devices [64]. Furthermore, the performed computational performances promote targeting future work for using higher resolution fundus images, in particular the ultra-wide field, and achieving real time implementations with higher segmentation accuracy.

Table 8
Computation requirements of the proposed model and state-of the art models.

Model	Params (M)	DRIVE	
		ACC	Time(s)
U-Net [39]	31.03	0.9554	-
D-U-Net[27]	0.88	0.9566	15.3
R-U-Net [29]	1.037	0.9556	6.42
R2U-Net[29]		0.9556	-
Our proposed RV-Net	**10.87**	**0.9819**	**0.59**

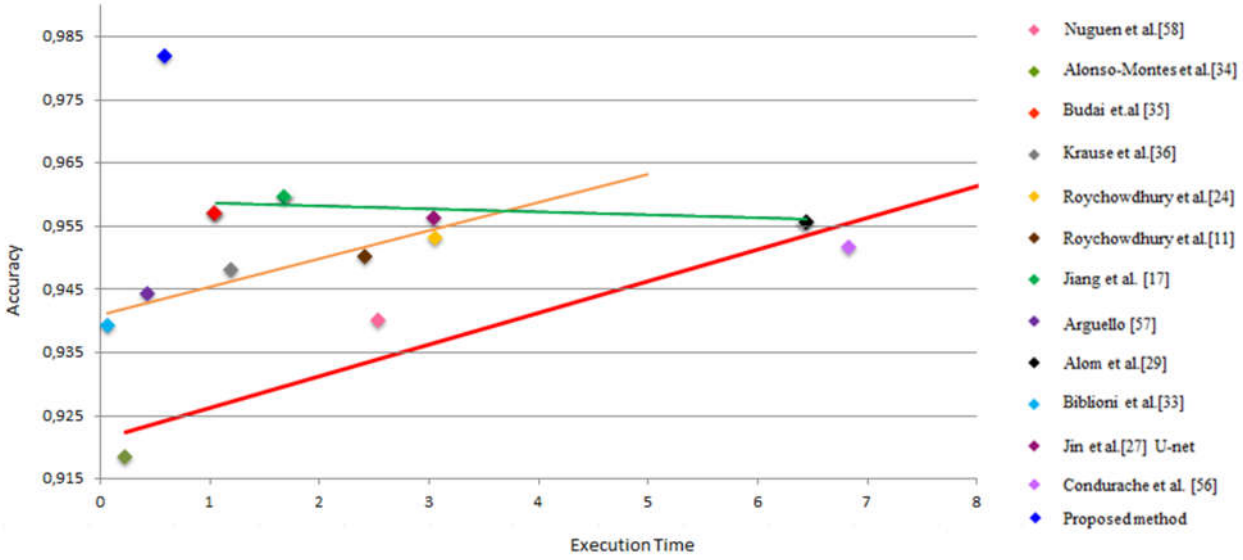


Fig. 13. Tradeoff between segmentation accuracy and execution time.

References

[1] S. Yu, D. Xiao, and Y. Kanagasingam, 'Automatic detection of neovascularization on optic disk region with feature extraction and support vector machine', in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Aug. 2016, pp. 1324–1327, doi: 10.1109/EMBC.2016.7590951.

[2] S. S. Kar and S. P. Maity, 'Detection of neovascularization in retinal images using mutual information maximization', *Computers & Electrical Engineering*, vol. 62, pp. 194–208, Aug. 2017, doi:

10.1016/j.compeleceng.2017.05.012.

[3] M.-C. C. Gicquel *et al.*, 'Indications et traitement de la dégénérescence maculaire liée à l'âge (DMLA) en pratique', *Sante Publique*, vol. Vol. 25, no. 3, pp. 315–324, Aug. 2013.

[4] S. Khitran, M. U. Akram, A. Usman, and U. Yasin, 'Automated system for the detection of hypertensive retinopathy', in *2014 4th International Conference on Image Processing Theory, Tools and Applications (IPTA)*, Oct. 2014, pp. 1–6, doi: 10.1109/IPTA.2014.7001984.

[5] W. M. D. W. Zaki, M. A. Zulkifley, A. Hussain, W. H. W. A. Halim, N. B. A. Mustafa, and L. S. Ting, 'Diabetic retinopathy assessment: Towards an automated system', *Biomedical Signal Processing and Control*, vol. 24, pp. 72–82, Feb. 2016, doi: 10.1016/j.bspc.2015.09.011.

[6] J. Kaur and D. Mittal, 'A generalized method for the segmentation of exudates from pathological retinal fundus images', *Biocybernetics and Biomedical Engineering*, vol. 38, no. 1, pp. 27–53, Jan. 2018, doi: 10.1016/j.bbe.2017.10.003.

[7] S. Moccia, E. De Momi, S. El Hadji, and L. S. Mattos, 'Blood vessel segmentation algorithms — Review of methods, datasets and evaluation metrics', *Computer Methods and Programs in Biomedicine*, vol. 158, pp. 71–91, May 2018, doi: 10.1016/j.cmpb.2018.02.001.

[8] M. M. Fraz *et al.*, 'Blood vessel segmentation methodologies in retinal images – A survey', *Computer Methods and Programs in Biomedicine*, vol. 108, no. 1, pp. 407–433, Oct. 2012, doi: 10.1016/j.cmpb.2012.03.009.

[9] K. B. Khan *et al.*, 'A review of retinal blood vessels extraction techniques: challenges, taxonomy, and future trends', *Pattern Anal Applic*, vol. 22, no. 3, pp. 767–802, Aug. 2019, doi: 10.1007/s10044-018-0754-8.

[10] J. Dash and N. Bhoi, 'A thresholding based technique to extract retinal blood vessels from fundus images', *Future Computing and Informatics Journal*, vol. 2, no. 2, pp. 103–109, Dec. 2017, doi: 10.1016/j.fcij.2017.10.001.

[11] S. Roychowdhury, D. D. Koozekanani, and K. K. Parhi, 'Iterative Vessel Segmentation of Fundus Images', *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 7, pp. 1738–1749, Jul. 2015, doi: 10.1109/TBME.2015.2403295.

[12] X. Xu *et al.*, 'Smartphone-Based Accurate Analysis of Retinal Vasculature towards Point-of-Care Diagnostics', *Sci Rep*, vol. 6, no. 1, pp. 1–9, Oct. 2016, doi: 10.1038/srep34603.

[13] A. Ali, A. Hussain, and W. M. D. W. Zaki, 'Vessel extraction in retinal images using automatic thresholding and Gabor Wavelet', in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Jul. 2017, pp. 365–368, doi: 10.1109/EMBC.2017.8036838.

[14] J. Odstrcilik *et al.*, 'Retinal vessel segmentation by improved matched filtering: evaluation on a new high-resolution fundus image database', *IET Image Processing*, vol. 7, no. 4, pp. 373–383, Jun. 2013, doi: 10.1049/iet-ipr.2012.0455.

[15] N. P. Singh and R. Srivastava, 'Retinal blood vessels segmentation by using Gumbel probability distribution function based matched filter', *Comput Methods Programs Biomed*, vol. 129, pp. 40–50, Jun. 2016, doi: 10.1016/j.cmpb.2016.03.001.

[16] R. Annunziata, A. Garzelli, L. Ballerini, A. Mecocci, and E. Trucco, 'Leveraging Multiscale Hessian-Based Enhancement With a Novel Exudate Inpainting Technique for Retinal Vessel Segmentation', *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 4, pp. 1129–1138, Jul. 2016, doi: 10.1109/JBHI.2015.2440091.

[17] Z. Jiang, J. Yepez, S. An, and S. Ko, 'Fast, accurate and robust retinal vessel segmentation system', *Biocybernetics and Biomedical Engineering*, vol. 37, no. 3, pp. 412–421, Jan. 2017, doi: 10.1016/j.bbe.2017.04.001.

[18] E. Imani, M. Javidi, and H.-R. Pourreza, 'Improvement of retinal blood vessel detection using morphological component analysis', *Computer Methods and Programs in Biomedicine*, vol. 118, no. 3, pp. 263–279, Mar. 2015, doi: 10.1016/j.cmpb.2015.01.004.

[19] B. Biswal, T. Pooja, and N. Bala Subrahmanyam, 'Robust retinal blood vessel segmentation using line detectors with multiple masks', *IET Image Processing*, vol. 12, no. 3, pp. 389–399, 2018, doi: 10.1049/iet-ipr.2017.0329.

[20] A. Christodoulidis, T. Hurtut, H. B. Tahar, and F. Cheriet, 'A multi-scale tensor voting approach for small retinal vessel segmentation in high resolution fundus images', *Comput Med Imaging Graph*, vol. 52, pp. 28–43, 2016, doi: 10.1016/j.compmedimag.2016.06.001.

[21] R. GeethaRamani and L. Balasubramanian, 'Retinal blood vessel segmentation employing image processing and data mining techniques for computerized retinal image analysis', *Biocybernetics and Biomedical Engineering*, vol. 36, no. 1, pp. 102–118, Jan. 2016, doi: 10.1016/j.bbe.2015.06.004.

[22] S. Aslani and H. Sarnel, 'A new supervised retinal vessel segmentation method based on robust hybrid features', *Biomedical Signal Processing and Control*, vol. 30, pp. 1–12, Sep. 2016, doi: 10.1016/j.bspc.2016.05.006.

[23] C. Zhu *et al.*, 'Retinal vessel segmentation in colour fundus images using Extreme Learning Machine', *Computerized Medical Imaging and Graphics*, vol. 55, pp. 68–77, Jan. 2017, doi: 10.1016/j.compmedimag.2016.05.004.

[24] S. Roychowdhury, D. D. Koozekanani, and K. K. Parhi, 'Blood Vessel Segmentation of Fundus Images by Major Vessel Extraction and Subimage Classification', *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 3, pp. 1118–1128, May 2015, doi: 10.1109/JBHI.2014.2335617.

[25] L. Zhang, M. Fisher, and W. Wang, 'Retinal vessel segmentation using multi-scale textons derived from keypoints', *Computerized Medical Imaging and Graphics*, vol. 45, pp. 47–56, Oct. 2015, doi: 10.1016/j.compmedimag.2015.07.006.

[26] Z. Jiang, H. Zhang, Y. Wang, and S.-B. Ko, 'Retinal blood vessel segmentation using fully convolutional network with transfer learning', *Computerized Medical Imaging and Graphics*, vol. 68, pp. 1–15, Sep. 2018, doi: 10.1016/j.compmedimag.2018.04.005.

[27] Q. Jin, Z. Meng, T. D. Pham, Q. Chen, L. Wei, and R. Su, 'DUNet: A deformable network for retinal vessel segmentation', *Knowledge-Based Systems*, vol. 178, pp. 149–162, Aug. 2019, doi: 10.1016/j.knosys.2019.04.025.

[28] C. Wang, Z. Zhao, Q. Ren, Y. Xu, and Y. Yu, 'Dense U-net Based on Patch-Based Learning for Retinal Vessel Segmentation', *Entropy*, vol. 21, p. 168, Feb. 2019, doi: 10.3390/e21020168.

[29] M. Z. Alom, M. Hasan, C. Yakopcic, T. M. Taha, and V. K. Asari, 'Recurrent Residual Convolutional Neural Network based on U-Net (R2U-Net) for Medical Image Segmentation', May 2018, *arXiv:1802.06955.*

[30] L. Luo, D. Chen, and D. Xue, 'Retinal blood vessels semantic segmentation method based on modified U-Net', in *2018 Chinese Control And Decision Conference (CCDC)*, Jun. 2018, pp. 1892–1895, doi: 10.1109/CCDC.2018.8407435.

[31] X. Xiao, S. Lian, Z. Luo, and S. Li, 'Weighted Res-UNet for High-Quality Retina Vessel Segmentation', in *2018 9th International Conference on Information Technology in Medicine and Education (ITME)*, Oct. 2018, pp. 327–331, doi: 10.1109/ITME.2018.00080.

[32] S. Sil Kar and S. P. Maity, 'Retinal blood vessel extraction using tunable bandpass filter and fuzzy conditional entropy', *Computer Methods and Programs in Biomedicine*, vol. 133, pp. 111–132, Sep. 2016, doi: 10.1016/j.cmpb.2016.05.015.

[33] P. Bibiloni, M. González-Hidalgo, and S. Massanet, 'A real-time fuzzy morphological algorithm for retinal vessel segmentation', *J Real-Time Image Proc*, vol. 16, no. 6, pp. 2337–2350, Dec. 2019, doi: 10.1007/s11554-018-0748-1.

[34] C. Alonso-Montes, D. L. Vilariño, P. Dudek, and M. G. Penedo, 'Fast retinal vessel tree extraction: A pixel parallel approach', *I. J. Circuit Theory and Applications*, 2008, doi: 10.1002/cta.512.

[35] A. Budai, R. Bock, A. Maier, J. Hornegger, and G. Michelson, 'Robust Vessel Segmentation in Fundus Images', *International Journal of Biomedical Imaging*, vol. 2013, p. 154860, Dec. 2013, doi: 10.1155/2013/154860.

[36] M. Krause, R. M. Alles, B. Burgeth, and J. Weickert, 'Fast retinal vessel analysis', *J Real-Time Image Proc*, vol. 11, no. 2, pp. 413–422, Feb. 2016, doi: 10.1007/s11554-013-0342-5.

[37] Y. Guo, Ü. Budak, and A. Şengür, 'A novel retinal vessel detection approach based on multiple deep convolution neural networks', *Computer Methods and Programs in Biomedicine*, vol. 167, pp. 43–48, Dec. 2018, doi: 10.1016/j.cmpb.2018.10.021.

[38] C. Tian, T. Fang, Y. Fan, and W. Wu, 'Multi-path convolutional neural network in fundus segmentation of blood vessels', *Biocybernetics and Biomedical Engineering*, vol. 40, no. 2, pp. 583–595, Apr. 2020, doi: 10.1016/j.bbe.2020.01.011.

[39] O. Ronneberger, P. Fischer, and T. Brox, 'U-Net: Convolutional Networks for Biomedical Image Segmentation', in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, 2015, pp. 234–241.

[40] P. Liskowski and K. Krawiec, 'Segmenting Retinal Blood Vessels With Deep Neural Networks', *IEEE Transactions on Medical Imaging*, vol. 35, no. 11, pp. 2369–2380, Nov. 2016, doi: 10.1109/TMI.2016.2546227.

[41] P. M. Samuel and T. Veeramalai, 'VSSC Net: Vessel Specific Skip

chain Convolutional Network for blood vessel segmentation', *Computer Methods and Programs in Biomedicine*, vol. 198, p. 105769, Jan. 2021, doi: 10.1016/j.cmpb.2020.105769.

[42] Q. Jin, Q. Chen, Z. Meng, B. Wang, and R. Su, 'Construction of Retinal Vessel Segmentation Models Based on Convolutional Neural Network', *Neural Processing Letters*, vol. 52, no. 2, pp. 1005–1022, Oct. 2020, doi: 10.1007/s11063-019-10011-1.

[43] A. Krizhevsky, I. Sutskever, and G. E. Hinton, 'ImageNet classification with deep convolutional neural networks', *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[44] K. Simonyan and A. Zisserman, 'Very Deep Convolutional Networks for Large-Scale Image Recognition', *arXiv:1409.1556 [cs]*, Apr. 2015, Accessed: Mar. 03, 2021. [Online]. Available: http://arxiv.org/abs/1409.1556.

[45] A. A. Mudassar and S. Butt, "Extraction of Blood Vessels in Retinal Images Using Four Different Techniques," *Journal of Medical Engineering*, vol. 2013, p. 408120, Dec. 2013, DOI: 10.1155/2013/408120.

[46] T. Chanwimaluang and Guoliang Fan, 'An efficient blood vessel detection algorithm for retinal images using local entropy thresholding', in *Proceedings of the 2003 International Symposium on Circuits and Systems, 2003. ISCAS '03.*, May 2003, vol. 5, p. V–V, doi: 10.1109/ISCAS.2003.1206162.

[47] R. J. Chalakkal and W. H. Abdulla, 'Improved Vessel Segmentation Using Curvelet Transform and Line Operators', in *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Nov. 2018, pp. 2041–2046, doi: 10.23919/APSIPA.2018.8659682.

[48] J. H. Tan *et al.*, 'Automated segmentation of exudates, haemorrhages, microaneurysms using single convolutional neural network', *Information Sciences*, vol. 420, pp. 66–76, Dec. 2017, doi: 10.1016/j.ins.2017.08.050.

[49] V. Couteaux *et al.*, 'Kidney cortex segmentation in 2D CT with U-Nets ensemble aggregation', *Diagnostic and Interventional Imaging*, vol. 100, no. 4, pp. 211–217, Apr. 2019, doi: 10.1016/j.diii.2019.03.001.

[50] S. Lian, Z. Luo, Z. Zhong, X. Lin, S. Su, and S. Li, 'Attention guided U-Net for accurate iris segmentation', *Journal of Visual Communication and Image Representation*, vol. 56, pp. 296–304, Oct. 2018, doi: 10.1016/j.jvcir.2018.10.001.

[51] A. G. Howard *et al.*, 'MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications', *arXiv:1704.04861 [cs]*, Apr. 2017, Accessed: Mar. 08, 2020. [Online]. Available: http://arxiv.org/abs/1704.04861.

[52] Y. Elloumi, M. Akil, and N. Kehtarnavaz, 'A mobile computer aided system for optic nerve head detection', *Computer Methods and Programs in Biomedicine*, vol. 162, pp. 139–148, Aug. 2018, doi: 10.1016/j.cmpb.2018.05.004.

[53] M. Abadi *et al.*, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," Mar. 2016, *arXiv:1603.04467*

[54] W. Wiharto and E. Suryani, 'The Comparison of Clustering Algorithms K-Means and Fuzzy C-Means for Segmentation Retinal Blood Vessels', *Acta Inform Med*, vol. 28, no. 1, pp. 42–47, Mar. 2020, doi: 10.5455/aim.2020.28.42-47.

[55] R. M. Heiberger and E. Neuwirth, 'One-Way ANOVA', in *R Through Excel: A Spreadsheet Interface for Statistics, Data Analysis, and Graphics*, R. M. Heiberger and E. Neuwirth, Eds. New York, NY: Springer, 2009, pp. 165–191.

[56] S. Wang, Y. Yin, G. Cao, B. Wei, Y. Zheng, and G. Yang, 'Hierarchical retinal blood vessel segmentation based on feature and ensemble learning', *Neurocomputing*, vol. 149, pp. 708–717, Feb. 2015, doi: 10.1016/j.neucom.2014.07.059.

[57] F. Argüello, D. L. Vilariño, D. B. Heras, and A. Nieto, 'GPU-based segmentation of retinal blood vessels', *J Real-Time Image Proc*, vol. 14, no. 4, pp. 773–782, Apr. 2018, doi: 10.1007/s11554-014-0469-z.

[58] U. T. V. Nguyen, A. Bhuiyan, L. A. F. Park, and K. Ramamohanarao, 'An effective retinal blood vessel segmentation method using multi-scale line detection', *Pattern Recognition*, vol. 46, no. 3, pp. 703–715, Mar. 2013, doi: 10.1016/j.patcog.2012.08.009.

[59] S. Yu, D. Xiao, and Y. Kanagasingam, 'Machine Learning Based Automatic Neovascularization Detection on Optic Disc Region', *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 3, pp. 886–894, May 2018, doi: 10.1109/JBHI.2017.2710201.

[60] A. Kerkeni, A. B. Abdallah, A. Manzanera, I. Nouira, and M. Bedoui, 'Automatic Bifurcation Detection in Coronary X-Ray Angiographies', p. 7.

[61] E. Sutanty, D. A. Rahayu, Rodiah, D. T. Susetianingtias, and S. Madenda, 'Retinal blood vessel segmentation and bifurcation detection using combined filters', in *2017 3rd International Conference on Science in Information Technology (ICSITech)*, Oct. 2017, pp. 563–567, doi: 10.1109/ICSITech.2017.8257176.

[62] D. Onkaew, R. Turior, B. Uyyanonvara, N. Akinori, and C. Sinthanayothin, 'Automatic retinal vessel tortuosity measurement using curvature of improved chain code', in *International Conference on Electrical, Control and Computer Engineering 2011 (InECCE)*, Jun. 2011, pp. 183–186, doi: 10.1109/INECCE.2011.5953872.

[63] S.Jerald Jeba Kumar and M.Madheswaran, 'Automated Thickness Measurement of Retinal Blood Vessels for Implementation of Clinical Decision Support Systems in Diagnostic Diabetic Retinopathy', Apr. 2010, doi: 10.5281/zenodo.1058546.

[64] M. Akil and Y. Elloumi, 'Detection of retinal abnormalities using smartphone-captured fundus images: a survey', in *Real-Time Image Processing and Deep Learning 2019*, Baltimore, United States, May 2019, p. 21, doi: 10.1117/12.2519094.

[65] Y. Elloumi, M. Akil, et H. Boudegga, « Ocular diseases diagnosis in fundus images using a deep learning: approaches, tools and performance evaluation », *Real-Time Image Processing and Deep Learning 2019*, vol. 10996, p. 109960T, May 20182019, doi: 10.1117/12.2519098.

[66] Y. Elloumi et M. A. Sr, « Computationally efficient blood vessels segmentation in fundus image on shared memory parallel machines », *Real-Time Image and Video Processing 2018*, , vol. 10670, p. 1067008, May 2018, doi: 10.1117/12.2304740