Received: 9 May 2019 | Revised: 2 October 2019 | Accepted: 3 February 2020

DOI: 10.1002/cpe.5719

RESEARCH ARTICLE

WILEY

Echo state network-based feature extraction for efficient color image segmentation

Abdelkerim Souahlia[1,2] | **Ammar Belatreche**[3] | **Abdelkader Benyettou**[1] | **Zoubir Ahmed-Foitih**[4] | **Elhadj Benkhelifa**[5] | **Kevin Curran**[6]

[1]Laboratoire Signal Image et Parole (SIMPA), Université des Sciences et de la Technologie d'Oran Mohamed Boudiaf, USTO-MB, Oran, Algeria

[2]Faculty of Sciences and Technology, University of Ziane Achour, Djelfa, Algeria

[3]Department of Computer and Information Sciences, Northumbria University, Newcastle-upon-Tyne, UK

[4]Laboratory of Power, Systems, Solar Energy and Automation LEPESA, Université des Sciences et de la Technologie, Mohamed-Boudiaf, USTO-MB, Bir El Djir, Algeria

[5]Cloud Computing and Applications Research Lab, Staffordshaire University, Stoke-on-Trent, UK

[6]School of Computing, Engineering and Intelligent Systems, Ulster University, Londonderry, UK

Correspondence
Abdelkerim Souahlia, Laboratoire Signal Image et Parole (SIMPA), Université des Sciences et de la Technologie d'Oran Mohamed Boudiaf, USTO-MB, BP 1505, El M'naouer, Oran 31000, Algérie.
Email: abdelkerim.souahlia@univ-usto.dz

Summary

Image segmentation plays a crucial role in many image processing and understanding applications. Despite the huge number of proposed image segmentation techniques, accurate segmentation remains a significant challenge in image analysis. This article investigates the viability of using echo state network (ESN), a biologically inspired recurrent neural network, as features extractor for efficient color image segmentation. First, an ensemble of initial pixel features is extracted from the original images and injected into the ESN reservoir. Second, the internal activations of the reservoir neurons are used as new pixel features. Third, the new features are classified using a feed forward neural network as a readout layer for the ESN. The quality of the pixel features produced by the ESN is evaluated through extensive series of experiments conducted on real world image datasets. The optimal operating range of different ESN setup parameters for producing competitive quality features is identified. The performance of the proposed ESN-based framework is also evaluated on a domain-specific application, namely, blood vessel segmentation in retinal images where experiments are conducted on the widely used digital retinal images for vessel extraction (DRIVE) dataset. The obtained results demonstrate that the proposed method outperforms state-of-the-art general segmentation techniques in terms of performance with an *F*-score of 0.92 ± 0.003 on the segmentation evaluation dataset. In addition, the proposed method achieves a comparable segmentation accuracy (0.9470) comparing with reported techniques of segmentation of blood vessels in images of retina and outperform them in terms of processing time. The average time required by our technique to segment one retinal image from DRIVE dataset is 8 seconds. Furthermore, empirically derived guidelines are proposed for adequately setting the ESN parameters for effective color image segmentation.

KEYWORDS

blood vessel segmentation, color image segmentation, echo state network, feature extraction, pixel classification, retinal images

1 | INTRODUCTION

Image segmentation consists of splitting an image into several disjoint regions which contain similar pixel features. Image segmentation is a crucial step in image understanding and analysis systems. Recently, many segmentation techniques have been reported in literatures,[1-4] yet accurate image

segmentation remains one of the important challenges in image analysis. That is mainly due to the similarities between different objects intensities, presence of noise, and poor contrast.

In recent years, a new paradigm of recurrent neural networks, namely, reservoir computing (RC), has been proposed. Its main purpose is to facilitate the task of recurrent neural networks training.[5,6] It has two models: the echo state network (ESN) proposed by Jaeger[7] and the liquid state machine (LSM) proposed by Maass et al.[8] The ESN model is composed of a large randomly generated untrained recurrent network of rate-based neurons and a readout layer. The latter represents the only trained part of the ESN. Linear regression algorithms are usually used to train the ESN readout layer. The LSM has a similar structure as the ESN model, however, it is based on spiking neuron models instead of rate-based neurons. Distinguished by the simplicity of its nodes and the ease of its training, the ESN has been used in many engineering applications.[9-13] Despite its simple architecture and ease of implementation, ESN configuration requires some practice and insight to obtain a good performance in many applications.[14] Several studies have been carried out to explore the ESN parameters and evaluate their performance in many engineering tasks.[15-17] However, in spite of the application of ESN to image segmentation in previous works,[12,13] to the best of our knowledge, a thorough investigation of its applicability to color image segmentation is still lacking. In addition, most of the applications based on ESN process a temporal data.[18,19] In fact, the main area of ESN success lies in the time series prediction.[9,20,21] However, in this work the ESN is used as a feature extractor which transforms input features into a new space where the newly extracted features become more easily separable.

In this work, we explore the viability of the ESN framework for color image segmentation. First, we extract an ensemble of pixel features from the image to be used as an input to our framework. Then, initial features are injected into the ESN reservoir. The reservoir acts as image feature extractor where the new features are represented by the reservoir internal nodes activations. Later, a simple feedforward neural network is used as a readout layer of the ESN to classify these new features. Extensive series of experiments on several real world image datasets have been conducted to thoroughly explore the ESN parameters and to examine the quality of image features extracted by the ESN reservoir. As a result, the operating ranges of the ESN parameters for obtaining high quality color image segmentation are then identified. Other series of experiments have also been conducted so as to evaluate the proposed ESN-based framework in domain-specific image segmentation, namely, segmentation of blood vessels in retinal images. This kind of segmentation has a significant role in automatic detection of some retinal diseases such as arteriosclerotic retinopathy, diabetic retinopathy, and hypertensive retinopathy.[22] Objective segmentation evaluation is used to assess the segmentation quality of the proposed ESN-based framework. Resulting segmentations are compared against the corresponding expert manual counterparts where several evaluation metrics are used to compute the segmentation performance. The obtained results demonstrate the effectiveness of the proposed framework for the segmentation of color images. Another finding in this study is that the use of only a small subset of randomly selected neurons from the ESN reservoir outputs proved sufficient to produce good quality image features which result in accurate segmentation.

The remainder of the article is organized as follows. A review of related literature is presented in Section 2. Section 3 presents the ESN model. Section 4 explains the application of the proposed framework to the segmentation of color images. Section 5 presents the experimental setup and describes the image datasets as well as the utilized simple image features, the evaluation metrics and the ESN framework setup. Section 6 reports and discusses the obtained results. Finally, conclusions are drawn in Section 7.

2 | RELATED WORK

In this section, we review existing relevant work and discuss different categories of image segmentation techniques. In addition, since we have chosen the blood vessel segmentation in retinal images as a case study, a review of relevant supervised techniques used in this vital domain-specific image segmentation is presented.

Lately, the ESN has been used as features extractors for classification and clustering tasks in several works. Most of them treated time signals such as time series and EEG signals. In Reference 23, authors have used ESN to discriminate positive and negative human emotions through the brain activities of multiple subjects. First, the initial vector of features extracted from EEG signals is mapped into the ESN reservoir. Then, the activations of the reservoir nodes are used as a new set of features. The authors have tried to extract the dominant features from different combinations of reservoir nodes. The final features are classified using several classifiers such as fuzzy C-means, K-means, K-nearest neighbors (KNNs), linear discriminant analysis, support vector machines, Naïve Bayes, and decision tree. In Reference 24, Sun et al proposed a method named feature extraction based on echo state network (FE-ESN). They have used an ESN as an autoencoder to realize a fully data-driven EEG feature extraction from multivariate EEG signals. The ESN mapped the EEG signals into a set of features represented by the states of the ESN reservoir nodes and then decoded them to recover the original EEG signals. The activations of the ESN nodes, which were used as a hidden layer of the autoencoder, are now used as new features for classification and clustering tasks. Experiments on real-world EEG datasets have been conducted where the authors demonstrated the effectiveness of FE-ESN comparing with the state-of-the-art techniques. In Reference 25, authors used a multilayer echo state network recurrent auto encoder for features extraction. In the proposed technique the capabilities of the ESN and those of the AE are mixed in an ESN-RAE framework to obtain efficient feature extraction. The newly obtained features lead to an increase in the classification accuracy achieved when solely using the original basic features. Most works based on ESN have treated temporal tasks such as time series and EEG signals classification. The main

area of ESN success is the time series prediction and classification. However, in this work the ESN is used for color images segmentation which is a static problem.

Image segmentation techniques can be divided into six classes: segmentation based on thresholding, segmentation based on regions, segmentation based on edges, segmentation based on clustering, segmentation based on graphs, and segmentation based on classification of pixels:

Thresholding-based segmentation is a one of the most popular image segmentation techniques and is based on image binarization using a threshold, hence their name.[26] Usually these techniques consist of three steps. First, the image histogram is computed. Second, the histogram is analyzed to extract the different modes and identify their valleys. Finally, appropriate thresholds are then applied to the image according to the identified valleys. Thresholding-based segmentation techniques are relatively simple, however, in the case of images with unimodal or nearly unimodal histograms, they do not work well. In addition, these techniques are not robust to noise. The presence of noisy peaks causes the appearance of fake modes which result in producing many ambiguous regions. In addition, thresholding is not a trivial task particularly in the case of color images having a multidimensional space.[27]

Region-based segmentation consists of separating an image into homogeneous regions where each region pixels are considered similar according to a predefined homogeneity criterion. Region splitting and merging and region growing are two common types of image segmentation based on regions. The major advantages of region-based segmentation techniques lie in their simplicity and their robustness to noise. However, they remain computationally excessive and often produce unclear objects boundaries.[28,29]

Edge-based segmentation techniques consist of detecting discontinuities between different image regions. The discontinuities or edges are the pixels corresponding to an intensity or texture abrupt change. Edge detection techniques suffer from high sensitivity to noise which results in detection of some fake edges and miss of some true ones.[30]

Clustering-based image segmentation is the process of separating the image pixels into different groups called clusters. This process takes into account two properties, the similarity between pixels of the same cluster and the dissimilarity between pixels belonging to different clusters. The most common algorithms of clustering are k-means[31] and fuzzy C-means.[32] While being simple to implement, techniques of clustering require a long processing time. In addition, the number of clusters in the image must be known in advance. The mean shift algorithm[33] is also a known clustering technique where pixels spatial coordinates (spatial domain) and features values (range domain) are considered together. For each pixel, the mean shift algorithm finds a stationary point by defining a window around the pixel, computing the mean of the pixels of this window, shifting its center to the computed mean and repeating these steps until finding the stationary point that corresponds to the center and the mean of pixels of the same window. All pixels corresponding to the same stationary point form a region. In this technique the prior knowledge of the clusters number is not required. In addition, this technique uses a few parameters only. However, it requires a model of the data distribution and the resulting segmentation depends on the window size.

Graph-based image segmentation is based on two steps. First, building a graph out of the image. Each pixel is considered as a vertex and its neighboring pixels are linked by an edge weighted by the distance between their features also called affinity or dissimilarity. Second, splitting this graph into subgraphs by minimizing a cost associated with the cut. A lot of methods have been proposed in this category of segmentation. They differ in the employed similarity measures, the cut cost functions and the optimization techniques.[34,35] The minimum cut criterion[34] favors cutting small groups of isolated nodes due to the low cut cost achieved by the partition of such nodes. In Reference 35, a new graph-based image segmentation called "normalized cuts" was proposed. It considers the total dissimilarity between the different subgraphs and the total similarity within the subgraphs. Generally, the graph-based techniques are computationally complex.

Pixel classification-based segmentation approaches the segmentation of an image as a problem of classification where a classifier is used to accord a label to each pixel according to its features (eg, intensity and texture). The same label (ie, the class or the group to which the pixel belongs) will be assigned to pixels having similar features. A region in the resulting segmentation is the ensemble of connected pixels assigned to the same class. Usually pixel classification uses supervised techniques which require training of a classifier on a subset of the image pixels. The rest of image pixels unseen during the training phase is then classified using the trained classifier. It is also possible to train a classifier using a group of training images then use the trained classifier for segmenting new unseen images. A good example of the latter case is medical imaging such as segmentation of blood vessels in images of the retina where each pixel is assigned to vessel or nonvessel (background) classes. Image segmentation by supervised techniques-based pixel classification often results in good segmentation performance. Nevertheless, these techniques involve a training time and the results may depend on the classifier parameters initialization, for example, initial weights of a neural network.

Numerous blood vessel segmentation supervised techniques have been proposed. Recent surveys of those techniques can be found in References 36,37. In this work, we review the most common techniques that are relevant to our approach. Niemeijer et al[38] used a vector of 31 pixel features. It consists of the Gaussian and its first and second derivatives at five different scales, then they used a KNN technique to classify pixels into vessel and background classes. The main disadvantage of this technique is that it produces false classification of the pixels that belong to thin vessels and those which are located around the optic disc. Staal et al[39] assumed that vessels have elongated structure and proposed a ridge-based vessel detection technique which computes 27 features for each pixel. They used a sequential forward selection method to select features with the best class separability and used k-NN to classify them. The limitations of this technique are the classification of the central part of the vessel as background, and the low accuracy in case of pathological images. In Reference 40 Soares et al used a Gaussian mixture model (GMM) Bayesian

classifier to assign each pixel to a vessel or background classes. The pixel features were extracted using the Gabor wavelet transform which performs a multiscale analysis of the image with different scales and orientations. The technique achieves a good segmentation performance. However, it results in low segmentation accuracy for the images with nonuniform illumination where false detections are produced in the pixels of the optic disc area and pixels corresponding to hemorrhages and pathologies that engender high similarity between vessel and background pixels. Marin et al[41] built a vector of seven features by combining moment-invariant with gray-level features. They used a multilayer feedforward neural network for classification. This technique resulted in a good segmentation performance on multiple image datasets even if the used classifier is trained on only one dataset. However, the processing time of one image is relatively high, due to the numerous preprocessing operations needed before feature extraction. Fraz et al[42] applied a decision trees ensemble for training and classifying pixels. Features are extracted based on gradient operator, Gabor filter, line strength calculations and mathematical morphology operators. Random forest is a widely used classifier in many engineering applications.[43] In Reference 44, it has been used to classify a large pool of features containing heterogeneous context-aware features, namely, Weber's local descriptors and stroke width transform, in addition to other classical local features such as intensity-based features, vesselness and Gabor-based features. The segmentation performance of this technique is comparable with other state-of-the-art techniques. However, the computation of such large number of features is time consuming. Deep learning techniques have been widely employed in the field of machine learning.[45,46] Several techniques of blood vessel segmentation in retinal images based on deep learning have been proposed in literature.[22,47,48] Usually, these techniques result in an accurate segmentation. However, they require a large amount of training samples. In addition, the required time to segment one image using these techniques is relatively long.

The present work extends two previous works by the authors.[49,50] In Reference 49, the study of the effect of ESN parameters on color image segmentation performance was limited to a reduced set of the ESN reservoir parameters only, which are the density of connectivity between reservoir nodes, the spectral radius, and the reservoir size. However, this new work further extends the study of the influence of other parameters on the quality of the segmentation by including: the effect of the selected color space, the order in which inputs are selected, the input scaling and the number of selected nodes from the ESN reservoir (see Section 6.1). Furthermore, while the previous study[49] used only one dataset, namely, the semantic segmentation dataset (SSDS),[51] in the present work we have conducted a wide range of experiments using popular datasets such as the segmentation evaluation dataset (SED)[52] and the DRIVE[39] datasets. In Reference 50, the ESN-based framework has been applied for segmenting blood vessels in retinal images using the DRIVE dataset where 13 pixel features were used (the RGB chromatic values, the mean, the SD, the gradient magnitude and angle in addition to six Gabor features). However, in this study we limit the set of features to the first seven simple features where the Gabor features are discarded. Moreover, we propose the use of the HSV color space instead of RGB. As shown in Section 6.1.1, this color space exhibits the best segmentation performance for the DRIVE dataset. These introduced changes result in improvements of segmentation performance in terms of accuracy and processing time.

Despite the existence of numerous segmentation techniques, accurate image segmentation remains a challenge. Furthermore, most of the reported techniques in blood vessel segmentation in retinal images suffer from long computational time. That is usually due to the large number of extracted features and the applied series of pre- and post-processing operations. The proposed pixel classification-based segmentation technique using ESN framework uses only a few simple pixel features. We demonstrate that the proposed method can outperform state-of-the-art general segmentation techniques in terms of performance. In addition, it can achieve a comparable segmentation accuracy compared with reported techniques of segmentation of blood vessels in images of retina and outperform them in terms of processing time. The following section introduces the ESN model and discusses its parameters.

3 | ESN—BACKGOUND

The structure of a generic ESN, as described in Reference 7, is shown in Figure 1. It involves three layers: the input layer, the internal layer (often called dynamic reservoir), and the output layer (usually called readout layer). Random synaptic input connections W_{in} connect the input layer to the ESN dynamic reservoir neurons. The ESN reservoir contains a large amount of neurons N. Randomly weighted connections W_{int} sparsely connect the reservoir nodes. The readout layer consists of L output neurons and is connected to the reservoir nodes through weighted output connections denoted by W_{out}. Table 1 summarizes the notations used throughout the article.

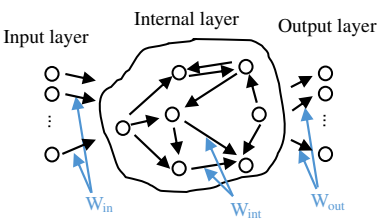

FIGURE 1 Architecture of a generic echo state network

TABLE 1 List of notations

Notation	Significance or Description
W_{in}	weights of input connections
W_{int}	Weights of reservoir nodes connections
W_{out}	Weights of output connections
N	Number of reservoir nodes
L	Number of output nodes
K	Number of input nodes
$x(n)$	Reservoir states at time step n
$y(n)$	Output of the ESN at time step n
$u(n)$	Input of the ESN at time step n
n	Time step
f	Activation function of the reservoir neurons
g	Activation function of the output nodes
Y_d	Target output vector
X	Matrix accumulating reservoir states
X^T	Transpose of X
$\|.\|$	Euclidean norm
$(X^TX)^{-1}$	Inverse of the matrix (X^TX)
R_i	Red channel value of the ith pixel
G_i	Green channel value of the ith pixel
B_i	Blue channel value of the ith pixel
M_i	Average value of the ith pixel
S_i	SD of the ith pixel
∇f	Gradient of a function f
g_x	Gradient in the x direction
g_y	Gradient in the y direction
$\partial f/\partial x$	Partial derivative of f with respect to x
$\partial f/\partial y$	Partial derivative of f with respect to y
G	Gradient magnitude
Θ	Gradient direction
arctg	Arctangent function
P	Precision
R	Recall
TP	True positives
TN	True negatives
FP	False positives
FN	False negatives
$\|\lambda_{max}\|$	Maximum absolute value of Eigen values
α	Spectral radius

The ESN dynamics are controlled by the following two equations:

$$x\,(n+1) = f\,(W_{\text{int}} \times x(n) + W_{\text{in}} \times u(n+1)) \tag{1}$$

$$y\,(n+1) = g(W_{\text{out}}\,x(n+1)) \tag{2}$$

where $x(n)$ and $x(n+1)$ are, respectively, the reservoir states at time steps n and $n+1$. $y(n+1)$ is the output of the ESN at time step $n+1$. The parameters f and g represent the activation functions of the reservoir neurons (usually a hyperbolic tangent, or any other sigmoidal function) and the output nodes (typically a linear function), respectively. $u(n+1) = \{u_j(n+1): j = 1,..., K\}$ is the input data at time step $n+1$. W_{in}, W_{int}, and W_{out} are, respectively, the matrices of weights for input nodes, reservoir nodes and output nodes. The sizes of W_{in}, W_{int}, and W_{out} are KxN, NxN, and NxL, respectively. K, N, and L are the number of nodes in the input layer, the reservoir and the output layer, respectively.

As mentioned early in the introduction, the ESN is distinguished mainly by its simple training procedure. The ESN training applies only to the reservoir-to-output connection weights W_{out}, whereas the weight matrices of the input and the reservoir, W_{in} and W_{int}, are randomly initialized and are then kept fixed. After feeding all the training input data to the ESN and computing the corresponding reservoir outputs, W_{out} is modified using the minimization of the mean squared error between the current (Y) and the target (Y_d) outputs:

$$W_{\text{out}} = \text{argw}(\min\|Y - Y_d\|^2) \tag{3}$$

where the symbol ||.|| indicates the Euclidean norm. Using linear regression, the readout weights matrix is given by Reference 7:

$$W_{\text{out}} = (X^T X)^{-1} X^T Y_d \tag{4}$$

where X is the matrix accumulating reservoir states, X^T the transpose of X and $(X^T X)^{-1}$ is the inverse of the matrix $(X^T X)$. However, in the present work, instead of using a single output layer, a multilayer perceptron (MLP) is used as a readout layer to classify the data collected from the reservoir states. In Reference 5, Lukosevicius et al report that the use of MLP as a readout layer of RC is theoretically more powerful in mappings from the reservoir state $x(n)$ to the ESN output $y(n)$ and appropriate for nonlinear outputs. In fact, Using MLP as RC readout layer dates back to the first appearance of LSM.[8]

The process of feeding of the input data $u(n)$ into the ESN and updating of the reservoir internal state $x(n)$ can be regarded as a projection of the input data ($u(n)$) into a higher dimensional space ($x(n)$). Usually, the original input data is not linearly separable. However, it is highly possible that the new data $x(n)$ can be easily separated if the reservoir parameters are tuned properly.[14] Therefore, the ESN performance is strongly affected by the configuration of the reservoir parameters. The following parameters of the ESN reservoir are studied in this article:

a. Neuron connectivity: This parameter defines the density of connection between the reservoir nodes. It is expressed by the nonzero elements distribution in W_{int}. A large amount of connections between reservoir neurons involves an increase in the number of operations required for calculating the reservoir state.

b. Reservoir size: Is the number of reservoir nodes. Usually big reservoirs are likely to find a linear combination of the signals to approximate the desired signal $Y_{d.}$[14] However, they increase the computational complexity. Consequently, in our experiments, we vary the reservoir size starting with smaller reservoirs then scaling up to higher dimensions.

c. Spectral radius: One of the key principles to ensure the applicability of ESN is the exhibition of echo state property (ESP) introduced by Reference 7. ESP means that the effect of reservoir parameters initialization vanishes after a limited time, that is, the state of the reservoir depends on the inputs and no longer on the ESN initial conditions. The spectral radius is a commonly used indicator of the ESP.[16] It is defined by the maximum absolute value of Eigen values $|\lambda_{\max}|$ of the matrix of the reservoir weights $W_{\text{int.}}$[17] Usually the ESP is ensured by setting the spectral radius to a value less than one. However, this common practice does not always succeed.[17] Therefore, an exploration of the operating range of these parameters is required.

In this study, the above-mentioned three parameters are explored and their effect on the segmentation performance is thoroughly investigated. The details of such investigation are presented in Section 6.1.5.

4 | THE PROPOSED ESN-BASED COLOR IMAGE SEGMENTATION APPROACH

In the present work, the segmentation based on the proposed framework is regarded as a problem of pixel classification. Each pixel is assigned to a class according to its features. First, an ensemble of initial features is extracted from the original images. These initial features are then fed into the ESN whose reservoir outputs are considered as new features of the pixels. Finally, using the labels extracted from the ground truth images, an MLP is trained to assign a label to each pixel according to its features. As shown in the flowchart presented in Figure 2, the steps of color image segmentation based on the proposed framework are:

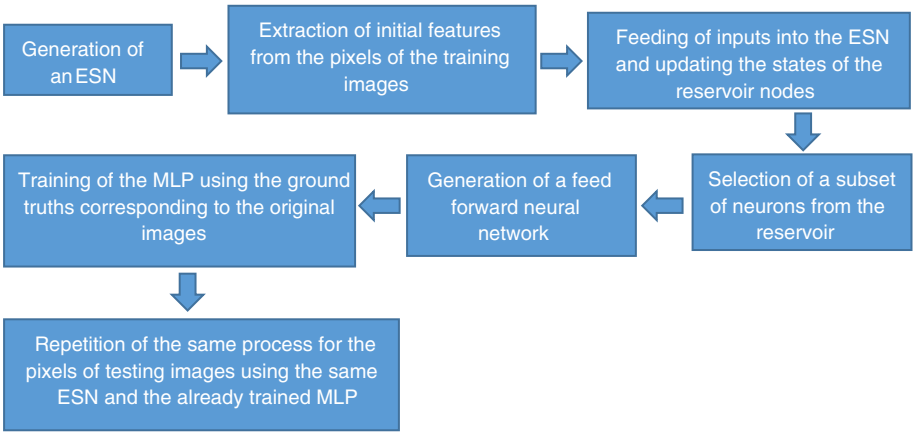

FIGURE 2 The proposed echo state network based framework for image segmentation

a. Generation of the ESN: an ESN having the ESP is generated as follows:

- A matrix of the weights of connections between reservoir nodes W_{int0} is randomly generated between −1 and + 1. W_{int0} should contain a number of zeros inversely proportional to the density of connectivity between the reservoir nodes.
- W_{int0} is normalized to W_{int1}:

$$W_{int1} = W_{int0}/|\lambda_{max}|, \quad \text{where } |\lambda_{max}| \text{ is the maximum absolute value of Eigen values of the matrix } W_{int0}. \tag{5}$$

- W_{int1} is scaled to W_{int}:

$$W_{int} = \alpha\, W_{int1} \tag{6}$$

where α is the spectral radius of W_{int}. Note that the ESP is strongly related to this parameter. A study of the influence of the spectral radius on the segmentation performance (along with other reservoir parameters such as the reservoir size and the density of connectivity between the reservoir nodes) is presented in Section 6.1.5.

- A matrix of weights of the input connections W_{in} is randomly generated between −1 and +1.

b. Extraction of the initial pixel features: for each pixel of the image, we have extracted five initial basic features $U_i = \{R_i, G_i, B_i, M_i, S_i\}$. R_i, G_i, B_i, are the three channels of the RGB color space and M_i and S_i are, respectively, the average and the SD of the neighboring pixels within a particular window. The window size can be chosen by trial and error. The feature extraction process is explained in more details in Section 5.2. Note that the idea behind the choice of simple initial pixels features as inputs for our proposed framework is to emphasize the ability of the ESN reservoir to extract a new set of good quality image features, which can achieve a good segmentation performance.

c. Feeding of inputs into the ESN: Each input U_i is projected onto the ESN reservoir and the corresponding reservoir output X_i is computed using Equation (1), that is, a further ensemble of pixels features exhibited by the activations of the neurons of the ESN reservoir is generated. For each pixel, the output of the reservoir is a vector containing the individual outputs of all the reservoir nodes. So, X_i is a vector of N components where N is the reservoir size.

d. Selection of a subset of neurons from the reservoir: It was found that the use of only a subset of neurons from the reservoir is sufficient to obtain good pixel features capable of achieving good segmentation performance (see Section 6.1.2). As mentioned in the previous step, each input U_i, corresponding to the initial features of a given pixel, is replaced by the corresponding reservoir output X_i. The final features of the pixel are X'_i which is a vector of M components randomly selected from the N components of X_i. A study of the influence of the number of selected neurons from the ESN reservoir on the segmentation performance is presented in Section 6.1.2.

e. *Generation of a feed forward neural network:* An MLP is used as a readout layer of the ESN to classify the data instead of using a single output layer. Details on the MLP architecture is delivered in Section 5.5.

f. Training of the MLP: After computing the final features of each pixel of the input image X'_I, a subset of pixels are selected for the MLP training. The expert manual segmentation corresponding to the input image is used as a target to train the MLP. During the training phase, the connection weights are updated using the Levenberg-Marquardt backpropagation algorithm. Section 5.6 shows further details on the training process.

g. Test of the framework: The features of the test images pixels are extracted using the same ESN and are classified using the already trained MLP. A label is assigned to each pixel. The region in the segmented image is the ensemble of connected pixels having the same label.

5 | EXPERIMENTAL SETUP

In this section, we discuss some choices related to the image segmentation based on the proposed framework such as the image datasets, the pixel features and the metrics used to evaluate resulting segmentations. We also discuss choices related to the configuration of the reservoir nodes (including their connection pattern, their activation functions and their connections with the input and the output layers) as well as the configuration of the readout layer and the training process.

5.1 | Benchmark datasets

In order to study different ESN parameters and evaluate the performance of the image segmentation based on the proposed framework, we have conducted a series of experiments on two real world image datasets: the SSDS[51] and the SED.[53] In addition, the proposed framework is validated on a domain-specific application dataset, namely, the digital retinal images for vessel extraction dataset (DRIVE).[39]

The SSDS contains 100 images of 321×481 pixels selected from Berkeley Segmentation Dataset (BSDS500).[54] The SSDS dataset has been created by Li et al[51] to be used for semantic segmentation. In this dataset, the ground truth segmentations of BSDS500, having often 10 to 30 segments, are refined and simplified to have only two to eight segments. Sample images, their original ground segmentations from the BSDS dataset and the refined ground truth segmentations from the SSDS dataset are presented in Figure 3.

The SED dataset includes 100 images, each of them contains a single prominent object.[53] The aim of the segmentation of this dataset images is to separate the background region from the foreground with the latter covering the main foreground object as accurately as possible. Examples from this dataset images and their corresponding expert manual segmentations are presented in Figure 4.

The DRIVE dataset contains 40 images of retina.[39] The dimension of every image is 768×584 pixels with a circular field of view (FOV) having a diameter of about 540 pixels. Segmentation of these images aims to obtain the tree of blood vessels from the retina. This dataset has been divided into two subsets of 20 images each, one for training and the other for testing.[39] Figure 5 shows samples from the DRIVE dataset images along with their corresponding ground truths.

Unlike the above mentioned two datasets (SSDS and SED) which contain natural scenes with different objects in each image, images in this dataset contain the same object (namely, the retina). As a result, this particular property of this dataset should, in principle, make the training of our

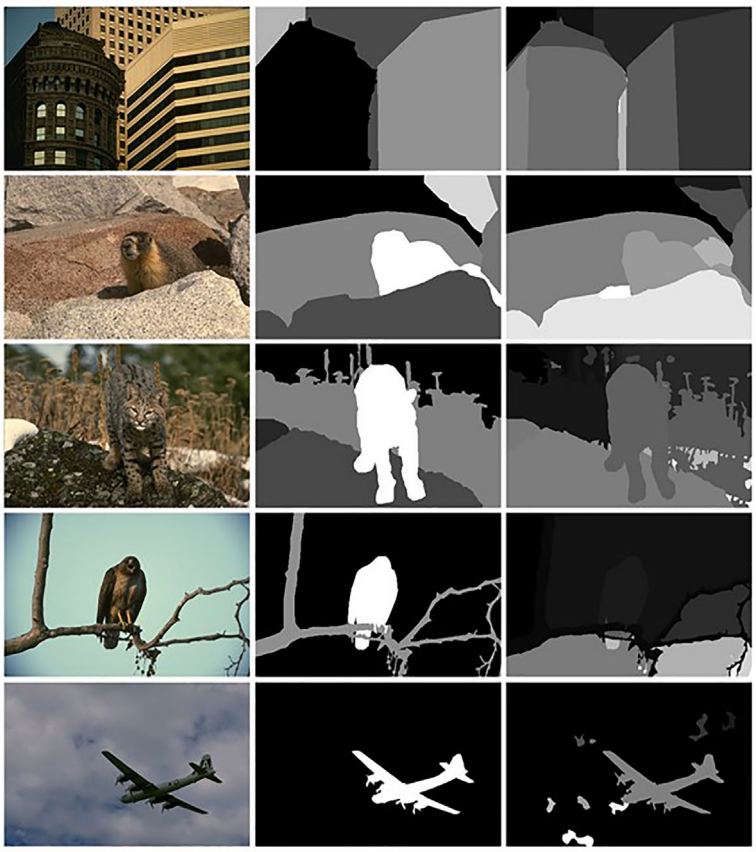

FIGURE 3 Sample images from the semantic segmentation dataset and the BSDS500 datasets. First column: Original images, second and third columns: expert manual segmentations from the semantic segmentation dataset and BSDS500 datasets, respectively

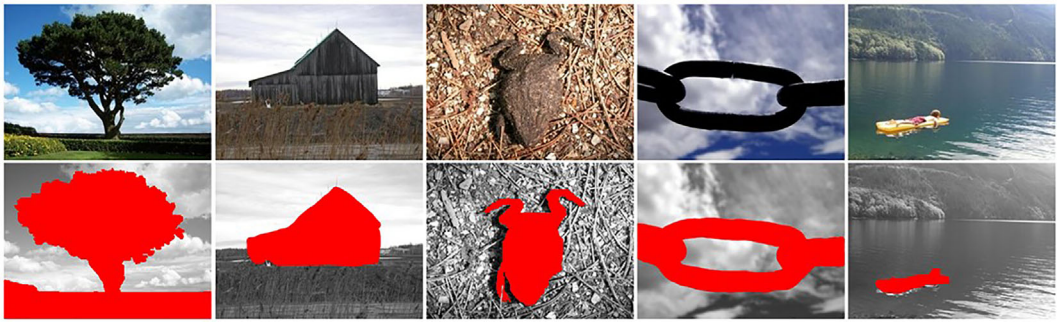

FIGURE 4 Sample images from the segmentation evaluation dataset with their ground truth segmentations

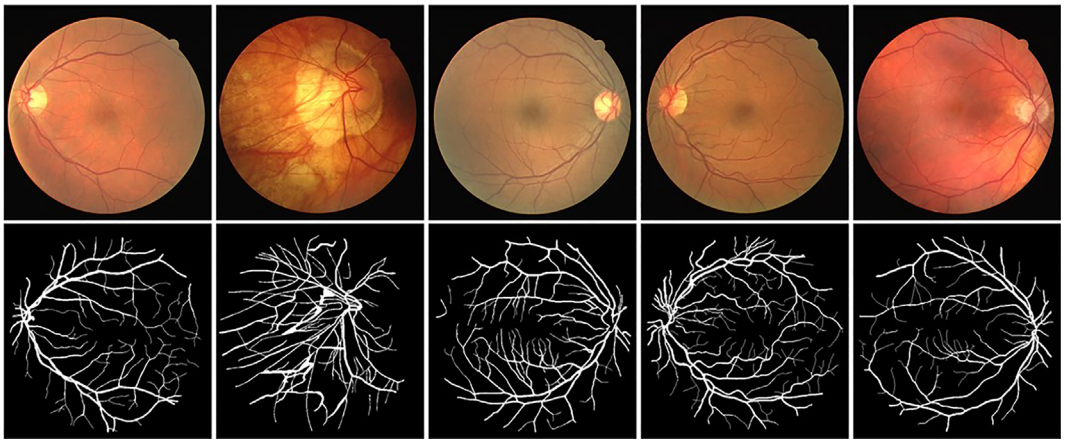

FIGURE 5 Sample images from the digital retinal images for vessel extraction dataset. Upper row: Original image. Lower row: corresponding ground truth

proposed framework less difficult than in the case of the other two datasets. We have used this dataset in order to further evaluate the viability of our ESN parameter design guidelines devised using the other two datasets, and to further compare our results with similar supervised techniques of pixel classification-based segmentation.

5.2 | Image features

In this work, a set of simple low level pixel features has been used as input of our proposed framework. These initial simple features have been fed into the ESN which extracts a further ensemble of features represented by the states of the reservoir nodes. The idea behind the choice of simple features is to emphasize the ability of the ESN reservoir to extract a new set of good quality image features which can lead to achieving a good segmentation performance. Therefore, in all our experiments conducted on the SSDS and SED datasets (which contains natural scenes), we have extracted five features. For each pixel, the three R, G, and B chromatic features are extracted. Note that the RGB color space has been chosen after conducting several experiments on several color spaces as shown later in Section 6.1.1. In addition, the SD and the mean values of each pixel and its adjacent pixels are used. They are computed through a window of 11×11 pixels. We have chosen this window size after testing several window sizes based on the trial and error principle. However, in our experiments on the DRIVE dataset containing retinal images, we have used for each pixel seven features. We have extracted the three HSV color space components as it has been found that this color space resulted in the best segmentation performance for this dataset (see Section 6.1.1). The mean and the SD within windows of 19×19 and 5×5 neighboring pixels, respectively, have been also extracted. These windows sizes are chosen through trial and error. In addition, and due to the complexity of distinguishing between vessel and background pixels using only the above-mentioned features, we have also used texture features based on gradient operator. We have extracted the texture features only from the green channel of the RGB color image without any preprocessing. As presented in Figure 6, the green plane (third column) has been found to exhibit the best vessel/background contrast whereas the red (second column), and blue (fourth column) channels exhibit

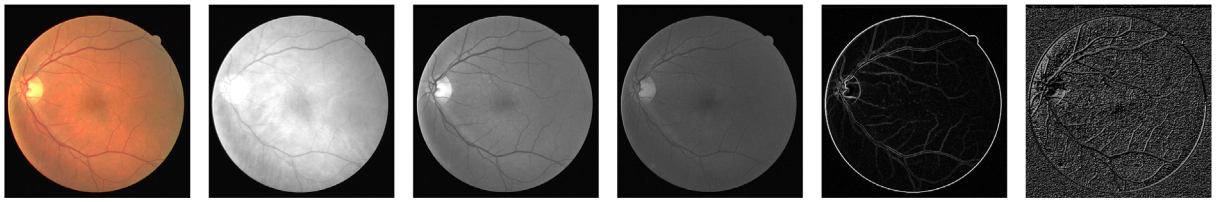

FIGURE 6 Example of the extracted features from the digital retinal images for vessel extraction dataset images. From left to right: original image, red channel, green channel, blue channel, gradient magnitude, and gradient direction

some noise.[41,44] It is worth noting that even with the use of these additional features, the feature vector remains simpler than those usually used by the other blood vessels segmentation techniques. A detailed explanation of this point will be presented in the section of results discussion.

5.2.1 | Gradient features

The gradient of an image is a directional change in the color or intensity in the image. It can be used to detect edges and extract texture features from images. The gradient at each image pixel is a two-dimensional (2D) vector composed by the derivatives in the vertical and horizontal directions. An image gradient can be computed using the following formula:

$$\nabla f = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \partial f/\partial x \\ \partial f/\partial y \end{bmatrix} \tag{7}$$

where: $\frac{\partial f}{\partial x}$ and $\frac{\partial f}{\partial y}$ are the gradients in the x and y directions, respectively. The most common practice to compute the gradient of an image is to calculate the convolution of the image with a kernel, such as the Sobel or the Prewitt operators. In our implementations, we have simply used the Prewitt operator which is defined as follows:

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -1 & 0 & +1 \\ -1 & 0 & +1 \end{bmatrix} * I \text{ and } G_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ +1 & +1 & +1 \end{bmatrix} * I \tag{8}$$

where * denotes the operation of 2D convolution and I is the source image. For each pixel of the image, the gradient magnitude G and direction Θ can be obtained by combining the vertical and horizontal gradient approximations:

$$G = \sqrt{G_x^2 + G_y^2} \tag{9}$$

$$\Theta = arctg(G_y/G_x) \tag{10}$$

Figure 6 (fifth column) and Figure 6 (sixth column) show an illustrative example of the gradient magnitude and direction, respectively, for an image from the DRIVE dataset.

Finally, all the used features in this work are normalized between 0 and 1. Note that each extracted feature (eg, mean, SD,…) is normalized separately.

5.3 | Evaluation metrics

In order to evaluate the segmentation accuracy based on the proposed framework, we have used several objective evaluation metrics:

5.3.1 | Accuracy, specificity, and sensitivity

For the experiments on the DRIVE dataset, we have evaluated the segmentation performance using the accuracy, sensitivity, and specificity metrics as they are commonly used in this field. The retinal image segmentation has two classes: vessel and nonvessel (background). Four measures are obtained by comparing the resulting segmentations with the expert manual segmentations: the true positives (T_P) which are vessels classified as vessels, false positives (F_P) which are nonvessels classified as vessels, true negatives (T_N) which are the nonvessels classified as nonvessels and the false negatives (F_N) which are the vessels classified as nonvessels. The accuracy, specificity, and sensitivity metrics are then computed using the

following equations:

$$\text{Accuracy} = \frac{T_P + T_N}{T_P + F_P + T_N + F_N} \tag{11}$$

$$\text{Specificity} = \frac{T_N}{T_N + F_P} \tag{12}$$

$$\text{Sensitivity} = \frac{T_P}{T_P + F_N} \tag{13}$$

5.3.2 | *F*-score

It is defined as follows:

$$F - \text{score} = 2 * P * \frac{R}{P + R} \tag{14}$$

P and *R* represent the precision and recall, respectively. The recall is equal to the sensitivity. The precision is defined by the following equation:

$$\text{Precision} = \frac{T_P}{T_P + F_p} \tag{15}$$

The *F*-score values range between 0 and 1. The higher the *F*-score value, the better the segmentation performance.

5.4 | ESN setup

All the input and the reservoir nodes connection weights are randomly initialized between −1 and +1. Hyperbolic tangent function has been used as a function of activation for the reservoir nodes. As we will see later, in Section 6.1.2, a subset of randomly selected neurons from the ESN reservoir is enough to extract good quality features for classification by the readout layer. Therefore, after conducting a number of experiments using different numbers of selected nodes from the ESN reservoir, 20 neurons have been chosen (see Section 6.1.2). Furthermore, the input layer does not have direct connections to the readout layer. In addition, the readout layer does not have feedback connections to the input layer or to the reservoir.

5.5 | MLP setup

As described early in this article, we have used an MLP as readout layer for the proposed ESN-based framework. According to the principle of trial and error, and after testing several configurations with various numbers of layers and various numbers of nodes in each layer, we have adopted an MLP with two hidden layers having 20 neurons in each. We have used only one output neuron for the DRIVE dataset having only two classes. The desired output of this neuron is 0 when the input pixel is part of the background and is 1 when the input pixel is a vessel. For the SED and the SSDS datasets which have two and from two to eight classes, respectively, we have used two output neurons. For the SED dataset, the desired outputs are $\{-1, -1\}$ for the foreground pixels and $\{1, 1\}$ for the background pixels. For the SSDS dataset, according to the number of classes present in the image, the desired outputs take values from the following couples: $\{-1, -1\}, \{1, 1\}, \{-1, 1\}, \{1, -1\}, \{-1, 0\}, \{0, -1\}, \{1, 0\}, \{0, 1\}$ (ie, for an image containing *n* classes, the first *n* couples are used as desired outputs, where *n* varies from 2 to 8).

5.6 | Training

In our experiments conducted on the SSDS and the SED datasets, we have randomly selected (without replacing) 40% of pixels from each image to train our readout. The remaining 60% of pixels are used for testing. Thus, for each image we have randomly selected from 15 000 to 55 800 and 61 760 pixels for SED and SSDS datasets, respectively. Note that the size of the SED dataset images varies from 125×300 to 465×300 pixels. As mentioned earlier, in Section 5.1, these two datasets contain natural images. Therefore, in all our experiments conducted on these two datasets, each image is processed separately. That is due to the fact that objects contained in these images vary from one image to another.

On the other hand, the DRIVE dataset contains retinal images where all of them contain the same objects which are vessels and background. This dataset is divided into 20 images for training and 20 images for testing.[39] Therefore, we have gathered the vessel pixels (569 415 pixels) and

nonvessel pixels (3 971 591 pixels) of all the 20 images of training. Afterward, we have randomly selected without replacing 56 941 vessel pixels, which is 10th of the total number of vessel pixels, and an equal number from the nonvessel pixels. Therefore, we have used a total of 113 882 pixels for training, which represents 2.51% of the total number of pixels in the training images. The remaining 20 testing images are used to test the segmentation performance based on the proposed framework.

6 | RESULTS AND DISCUSSION

In this section, we explore the influence of different parameters of the ESN on the performance of color image segmentation (Section 6.1). Such investigation will allow us to determine the operating ranges of these parameters, which helps in achieving high segmentation accuracy. In addition, a comparison of the performance of the proposed approach against that of other state-of-the-art techniques is conducted (Section 6.2).

6.1 | Study of the influence of ESN parameters on color image segmentation performance

In this section, we explore the influence of the following factors on the segmentation performance: the color space (Section 6.1.1), number of selected neurons from the reservoir (Section 6.1.2), pixel selection order (Section 6.1.3), input scaling (Section 6.1.4), and the global reservoir parameters covered in Section 3 (Section 6.1.5). Table 2 shows the studied parameters with their ranges of variation.

6.1.1 | Color space

Color image segmentation depends on the used color space. In this work, several experiments are conducted to find the color space that best suits the proposed framework. There exists several color spaces so we focus on the four most commonly used ones, namely, RGB, CIE-Lab, HSV, and YCbCr. RGB is defined by the main colors: red (R), green (G), and blue (B). It has remained the most basic color space used in image processing as the human color perception is based on this triplet of colors. HSV has three components: hue (H), saturation (S), and intensity value (V). The H takes values from 0 to 1, hence the colors change from red to yellow, green, blue, and black to red. While S takes values from 0 to 1, the hues change from unsaturated to fully saturated. V represents the brightness which varies from 0 (the darkest value) to 1 (the brightest value). The transformation between RGB and HSV is nonlinear. HIS, HLS, and HCI are the other similar color spaces to HSV. YCbCr contains a gray-scale information shown by the luminance component *Y*, and a color information represented by the two chrominance components Cb and Cr. A linear transformation associates the RGB and YCbCr color spaces. Other similar color spaces to YCbCr are YUV and YIQ. CIE-lab is a color space created by the international commission on illumination. The three components of CIE-Lab are the lightness "l" and the color channels "a" and "b." "l" varies from 0 (the black) to 100 (the white). "a" and "b" are the red/green and yellow/blue opponents, respectively. The transformation between RGB and CIE-Lab spaces is highly nonlinear. An example of a similar color space to CIE-Lab is the CIE-Luv.[55]

Table 3 shows the obtained average *F*-score using the selected four-color spaces. The ESN parameters are set as follows: the connectivity is set to 0.2, the reservoir size is set to 100 and the spectral radius is set to 0.1.

For the SED and the SSDS datasets, which contain natural scenes, the resulting averaged *F*-score using RGB slightly outperforms those of other color spaces. The averaged *F*-score corresponding to YCbCr comes in second place with a value very close to that corresponding to RGB. The third place alternated between CIE-Lab and HSV color spaces.

TABLE 2 Range of variation of different simulation parameters

Parameter	Variation Range
Color space	RGB, HSV, YCbCr, and CIE-Lab
Number of selected neurons from the reservoir	2, 5, 10, 20, 30, 40, 50, 60, 70, 80, 90, and 100
Order of input selection	Standard order, zigzag order, and random order
Input scaling	1e-20, 1e-15, 1e-10, 1e-5, 1e-3, 0.01, 0.1, 1, and 10
Spectral radius	0.001, 0.01, 0.1, 0.2, 0.4, 0.6, 0.8, and 1
Density of connectivity between reservoir nodes	Varied from 0.2 to 1 with a step of 0.2
Reservoir size	50, 100, 200, 300, and 500

TABLE 3 Averaged *F*-score of the proposed ESN-based approach for different color spaces

	Averaged *F*-Score		
Color Space	**SED Dataset**	**SSDS Dataset**	**DRIVE Dataset**
RGB	0.9125 ± 0.0013	0.8687 ± 0.0038	0.7418 ± 0.0049
HSV	0.9035 ± 0.0018	0.8658 ± 0.0043	0.7601 ± 0.0028
YCbCr	0.9110 ± 0.0032	0.8685 ± 0.0035	0.7578 ± 0.0047
CIE-Lab	0.9099 ± 0.0025	0.8597 ± 0.0046	0.7522 ± 0.0064

Abbreviations: DRIVE, digital retinal images for vessel extraction; ESN, echo state network; SED, segmentation evaluation dataset; SSDS, semantic segmentation dataset.

For the DRIVE dataset, which contains retinal images, the HSV color space slightly outperforms that of other color spaces. The RGB color space has resulted in the worst segmentation performance for this medical imaging application dataset. In addition, it is worth noting that for each dataset, the different values of *F*-score corresponding to different color spaces lie within a tight range. The *F*-score differences between the best and worst cases are 0.0090, 0.0090, and 0.0183 for the SED, the SSDS and the DRIVE datasets, respectively. Therefore, it is fair to conclude that the color space does not seem to have a huge impact on the segmentation performance. Consequently, the RGB color space is preferable for the SED and SSDS datasets as it is producing the best *F*-score performance. However, for the DRIVE dataset, the preferred color space is the HSV.

6.1.2 | Number of selected neurons from the reservoir

Image segmentation based on the proposed framework consists in projecting pixel features onto the ESN which maps them into a further ensemble of features shown by the states of the reservoir. This feature mapping is then followed up by an MLP which realizes the ESN readout function and classifies the new pixel features represented by the reservoir states. Usually, the whole ESN reservoir output is processed by the readout layer (MLP in our case), which is computationally expensive especially when the reservoir size is big. However, in this work only a subset of neurons is randomly selected from the reservoir rather than using all of the reservoir neurons. Figure 7 shows the evaluation of the effect of the number of selected neurons from the reservoir on the segmentation performance in terms of the *F*-score and the computation time. The connectivity between reservoir nodes is set to 0.2, the reservoir size is set to 100 and the spectral radius is set to 0.1. Experiments are conducted on all images of the SED, the SSDS, and the DRIVE datasets. Figure 7 illustrates the evolution of the average *F*-score (first row) and the average computation time (second row) against the number of randomly selected neurons from the reservoir. Each experiment was repeated five times for each number of selected neurons and the resulting SDs are illustrated by error bars. The computation time consists of training and testing times. Due to different initial conditions of the readout layer, the training time changes from one experiment to another even for the same number of selected nodes from the reservoir which is illustrated through higher error bars. For the *F*-score graphs, the SD is very small for all numbers of selected neurons from the reservoir in all conducted experiments on different datasets. The highest SD values are 0.0018 (corresponds to five neurons), 0.006 (corresponds to two neurons), and 0.0453 (corresponds to five neurons) for the SED, the SSDS and the DRIVE datasets, respectively. These small SDs clearly demonstrate the robustness of the proposed framework against random selection of the reservoir neurons. In addition, it can be seen that the *F*-score increases when the number of selected neurons increases for all the conducted experiments on the different datasets. However, the interesting remark is that the performance remains almost constant once the number of selected neurons reaches 20 for the SED and the DRIVE datasets and 40 for the SSDS dataset. This shows that using only 20% (for the SED and the DRIVE datasets) or 40% (for the SSDS dataset) of the reservoir neurons proved to be sufficient for achieving almost the same performance when using the entire reservoir. This is interesting as using only a subset of the reservoir neurons leads to a dramatic decrease in computation time. For instance, for the DRIVE dataset using only 20 neurons required a computation time of 582 seconds and achieved similar performance to using the entirety of the reservoir neurons which requires a computation time of 1603 seconds. That is, it is clearly possible to achieve similar segmentation performance by using only a subset of the reservoir neurons while at the same time reduce the computation time by nearly two-thirds. The experiments are conducted using MATLAB 2015a on an Intel i5-6400 CPU running at 2.7 GHz with 8 GB of RAM. On the basis of this finding, one can make an informed choice of the number of selected neurons from the reservoir that can provide a reasonable trade-off between adequate segmentation performance and computational time.

6.1.3 | Order of input selection

The ESN reservoir can be seen as a short-term memory that is, when an input $u(n)$ is fed into the reservoir, it is retained in its internal state $x(n)$ and is used to compute its new state $x(n+1)$ once a new input $u(n+1)$ is fed (see Equation 1). In order to study the effect of the order in which pixels

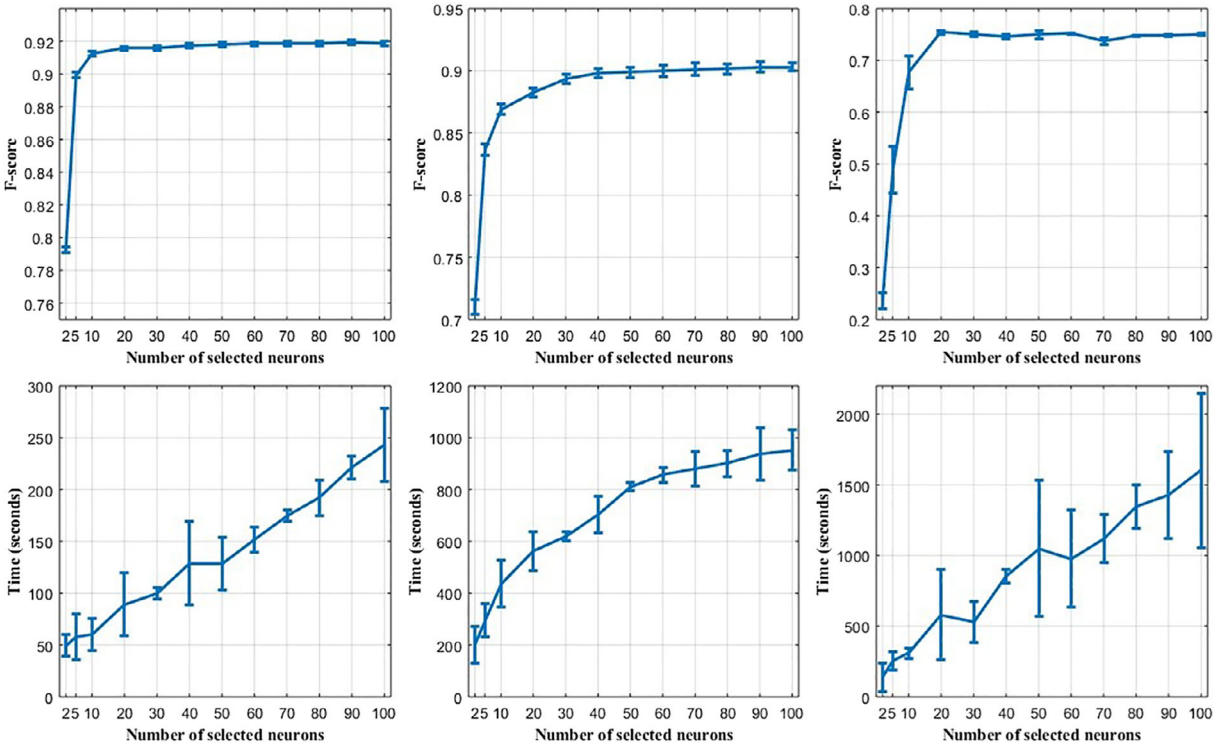

FIGURE 7 The effect of the number of selected neurons from the reservoir on the average *F*-score (upper row) and the processing time (lower row). Left column: the SED dataset. Middle column: the SSDS dataset. Right column: the DRIVE dataset. DRIVE, digital retinal images for vessel extraction; SED, segmentation evaluation dataset; SSDS, semantic segmentation dataset

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25
26	27	28	29	30

(A)

1	2	6	7	15
3	5	8	14	16
4	9	13	17	24
10	12	18	23	25
11	19	22	26	29
20	21	27	28	30

(B)

4	12	24	17	20
8	19	3	9	16
5	13	21	11	25
1	10	2	28	6
27	14	18	15	22
30	23	7	29	26

(C)

FIGURE 8 Different orders of pixel selection. (A) Standard order, (B) zigzag order, and (C) random order

TABLE 4 Averaged *F*-score of the proposed ESN-based approach for different orders of input pixel selection

	Averaged *F*-Score		
Order Type	**SED Dataset**	**SSDS Dataset**	**DRIVE Dataset**
Standard order	0.9159 ± 0.0012	0.8604 ± 0.0041	0.7611 ± 0.0032
Random order	0.9071 ± 0.0025	0.8308 ± 0.0034	0.7592 ± 0.0061
Zigzag order	0.9117 ± 0.0028	0.8501 ± 0.0054	0.7429 ± 0.0072

Abbreviations: DRIVE, digital retinal images for vessel extraction; ESN, echo state network; SED, segmentation evaluation dataset; SSDS, semantic segmentation dataset.

are selected, for each input image we have presented the image pixel features to the reservoir in three different orders: the standard order, which is line by line of the image pixels, the zigzag order, and the random order where pixels are randomly selected. Figure 8 shows how we scan a block of 6×5 pixels according to the three orders. The connectivity is set to 0.2, the reservoir size is set to 100 and the spectral radius is set to 0.1. Table 4 illustrates the average *F*-score obtained across all the images of the SED, the SSDS, and the DRIVE datasets for the three different pixel selection orders. It appears from the table that the pixel selection order does not have much influence on the segmentation accuracy. The standard order has a slightly better performance than the other two selection orders, followed by the zigzag order then the random order. Therefore, we will use the standard order in all our experiments as it is the simplest order in addition to resulting in the best performance.

6.1.4 | Input scaling

The input scaling is another aspect to explore when designing an ESN as it determines the degree of nonlinearity of the reservoir dynamics. Knowing that we have used a sigmoidal activation functions (in particular the hyperbolic tangent) for the reservoir nodes, a very small input scaling value makes the reservoir behave almost like a linear medium because the reservoir nodes operate around the zero point where their sigmoid activations are linear. However, large input scaling values drive the neurons to the saturation of the sigmoid (close to their −1 and +1 limit values).[14] In this section, experiments are carried out to investigate the effect of the input scaling on the segmentation performance of the proposed ESN-based framework. The spectral radius is set to 0.1, the reservoir size is set to 100 nodes, the connectivity between the reservoir nodes is set to 0.2, and the input scaling is set to the following values: 1e-20, 1e-15, 1e-10, 1e-5, 1e-3, 0.01, 0.1, 1, and 10. Figure 9 shows the evolution of the average *F*-score in terms of the input scaling for the SED, the SSDS and the DRIVE datasets. For the three datasets, the averaged *F*-score is almost stable for an input scaling comprised between 1e-10 and 1. However, very small values of input scaling can result in a low segmentation performance as in the case of the SSDS dataset shown in the middle column of Figure 9. On the other hand, an input scaling greater than 1 can result in a decrease in the averaged *F*-score as in the case of the DRIVE dataset shown in the right column of Figure 9. Note that the operating range of the input scaling parameter for the SED dataset is larger than those for the SSDS and DRIVE datasets. That is due to the simplicity of the process of segmentation of the SED dataset images comparing with the other two datasets images. The best performance over all input scaling values for the SED dataset (0.9158) is higher than those for the SSDS and DRIVE datasets (0.8651 and 0.7701, respectively). In fact, each image in the SED dataset contains only two objects. However, SSDS images contain two to eight objects. DRIVE dataset images contain also two objects, which are the vessels and the background, however, these objects have high overlaps between them. In general, the averaged *F*-score is almost stable and acceptable for an input scaling comprised between 1e-10 and 1 for the three datasets.

6.1.5 | Reservoir parameters

It is equally important to investigate the influence of the reservoir parameters (reservoir size, connectivity, and spectral radius) on the segmentation performance of the proposed ESN-based framework. These parameters are described earlier in Section 3. Such investigation will allow us to find the optimal operating ranges of these parameters and to derive guidelines for designing ESN reservoir for color image segmentation. In this section, an extensive series of experiments are carried out to study the effect of the three reservoir parameters on the ESN-based segmentation performance. Every time we vary these three reservoir parameters, we evaluate the segmentation performance of the resulting ESN. For each ESN configuration, the readout layer is trained and tested on all images of the three datasets (SED, SSDS, and DRIVE). The reservoir parameters are varied as follows: the spectral radius is set to 0.001, 0.01, 0.1, 0.2, 0.4, 0.6, 0.8, and 1; the density of connectivity between reservoir nodes is varied from 0.2 to 1 with a step of 0.2; and the reservoir size is set to 50, 100, 200, 300, and 500. Each experiment corresponding to a triplet of the three ESN reservoir parameters is repeated five times and the mean *F*-score is computed. For the SED and the SSDS datasets, each of which containing 100 images, we have conducted 100 000 experiments on each dataset. That is the number of possible triplets of the reservoir parameters values (5*8*5 = 200) by the number of dataset images (100) by the number of repetition of each experiment (5). For the DRIVE dataset, we have conducted 1000 experiments. For each ESN configuration, we have trained our framework using 20 training images and tested its performance using the remaining 20 testing images. Each experiment is repeated five times. Therefore, the total number of conducted experiments on all datasets is 201 000. Usually, cloud

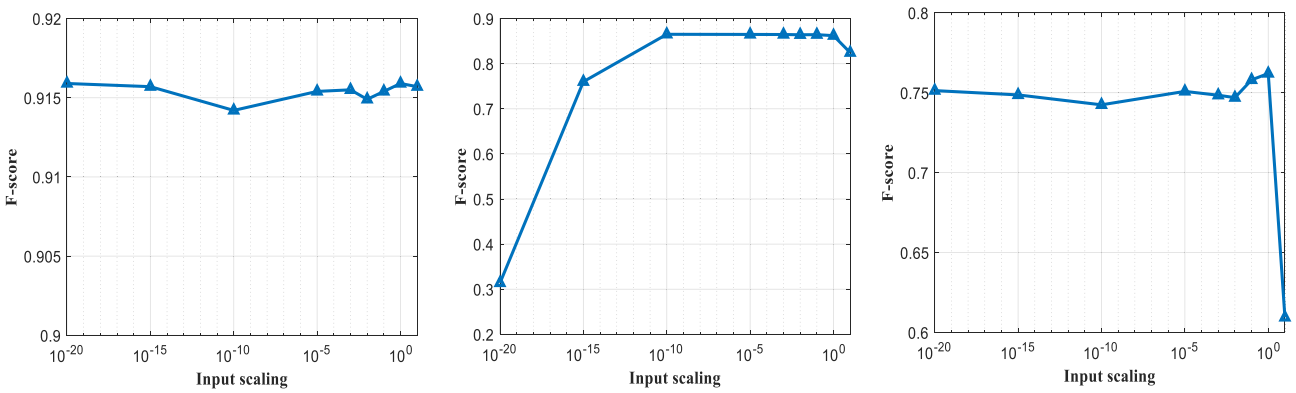

FIGURE 9 The effect of the input scaling on the average *F*-score. The logarithmic scale is used for the *x*-axis. Left column: the SED dataset, middle column: the SSDS dataset and right column: the DRIVE dataset. DRIVE, digital retinal images for vessel extraction; SED, segmentation evaluation dataset; SSDS, semantic segmentation dataset

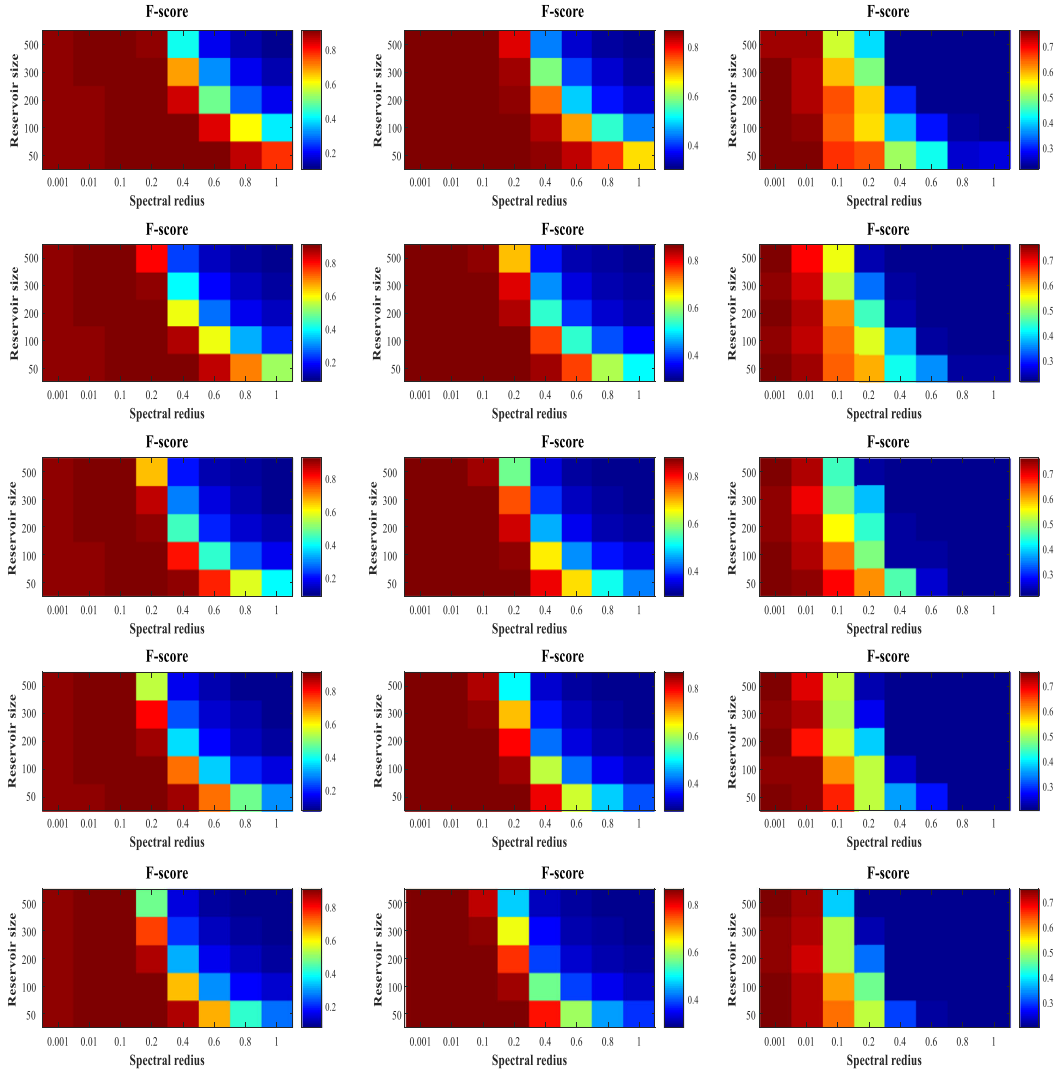

FIGURE 10 The average *F*-score as a function of the spectral radius and the reservoir size for different values of the connectivity density. From left to right columns: SED, SSDS, and DRIVE datasets; from upper to lower rows connectivity: 0.2, 0.4, 0.6, 0.8, and 1. DRIVE, digital retinal images for vessel extraction; SED, segmentation evaluation dataset; SSDS, semantic segmentation dataset

computing and high-performance computing (HPC) platforms are used for running such computationally intensive tasks.[56] In our work, we have used a HPC cluster, which consists of 32 × Dell R410s, each comprising of 2 × 6 core CPUs and 24 GB of RAM. Giving a total of 384 Cores and 768 GB of RAM. It operates on Windows Server 2008 R2 HPC Edition and uses Matlab 2013a.

Columns (A), (B), and (C) of Figure 10 show the mean *F*-score in function of the reservoir parameters values using the SED, the SSDS, and the DRIVE datasets, respectively. Each panel presents the mean *F*-score in terms of the spectral radius and the reservoir size for a given value of the connectivity density. Figure 10 shows that the overall performance results are comparable across all panels of each dataset regardless of the connectivity density value. Therefore, the density of connectivity does not seem to highly affect the performance of segmentation. However, ESN reservoirs with sparse connections between neurons are less complex than those fully connected ones. Thus, small density of connectivity values is preferable. It can also be seen from Figure 10 that increasing the spectral radius value up to 0.2 for SED and SSDS datasets (and up to 0.01 for DRIVE dataset) seems to give the best segmentation performances. However, increasing the spectral radius beyond these limits starts to cause a decrease in the segmentation performance. This decrease is more pronounced when the reservoir dimension and density connectivity are increased. The operating ranges of reservoir parameters for the SED and SSDS datasets are larger than those for the DRIVE dataset. That is due to the complexity of distinguishing between vessels and background pixels because of the high overlaps between them. This statement is also supported by the small best performance over all ESN reservoir parameters values for the DRIVE dataset comparing with those of the SED and SSDS datasets. These performances are 0.9159, 0.8687, and 0.7650 for SED, SSDS, and DRIVE datasets, respectively.

TABLE 5 Comparison of the proposed ESN-based framework segmentation against other state-of-the-art techniques in terms of the mean *F*-score using the SED dataset

Technique	Mean *F*-Measure
ESN	**0.92 ± 0.003**
Scheme[57]	0.87 ± 0.010
Scheme[53]	0.86 ± 0.012
SWA[58]	0.83 ± 0.016
Normalized cuts[35]	0.72 ± 0.018
Mean shift[33]	0.57 ± 0.023

Abbreviations: ESN, echo state network; SED, segmentation evaluation dataset.

In summary, based on the above extensive experimental evaluation conducted on the three image datasets (SED, SSDS, and DRIVE), the best design choice of the proposed framework parameters for color image segmentation can be summarized as follows:

- a use of the RGB color space for natural images and the HSV color space for retinal images
- a selection of only 20% of nodes from the reservoir can be used to extract good quality pixel features
- a selection of input image pixels in standard order is sufficient
- an input scaling comprised between 1e-10 and 1
- a reservoir with a spectral radius less than or equal to 0.2 for natural images and up to 0.01 for retinal images, a density of connectivity of 0.2 and a reservoir size between 50 and 100 nodes.

Consequently, these experimentally derived guidelines are used to compare the proposed system with other state-of-the-art image segmentation techniques in the following section.

6.2 | Comparison against state-of-the-art techniques

In this section, we compare the results of segmentation based on the proposed framework against state-of-the-art techniques. First, a comparison against general segmentation techniques using natural scenes is presented in Section 6.2.1. Then a comparison against supervised techniques using retinal images is presented in Section 6.2.2.

6.2.1 | Comparison using natural scenes

In this section, we conduct a comparison of the ESN-based color image segmentation performance with some state-of-the-art image segmentation techniques including the well-known mean shift and normalized cuts techniques.[33,35,53,57,58] All the techniques are evaluated on the SED dataset images. The obtained *F*-scores by the other techniques and the SED dataset images are available online.[59] Table 5 reports the averaged *F*-score and the SD. Based on the results presented in the past subsections, the ESN reservoir parameters are set as follows: the density of connectivity between the reservoir nodes is set to 0.2, the input scaling is set to 1, the spectral radius is set to 0.1 and the reservoir size is set to 100 nodes. In addition, we have randomly selected 20 reservoir nodes based on the results presented in Section 6.1.2. It is known that the quality of pixel features has a major influence on the segmentation performance. Using the found operating range of the ESN parameters, the proposed framework has allowed us to obtain good pixel features. These features, represented by the ESN reservoir outputs, facilitate the task of separation between the foreground object and the background. As a result, the proposed ESN-based segmentation approach outperformed the other state-of-the-art techniques as presented in Table 5.

In addition, Figure 11 shows sample segmentations from the SED dataset using the proposed ESN-based framework and those of state-of-the-art techniques. It can be clearly seen that the proposed technique, unlike other techniques, results in a segmentation that is very close to the expert manual segmentations. However, false detections can be produced in pixels that have different characteristics compared with their neighboring pixels. Thus, our framework is not robust to outlier pixels. For example, in our segmentation of the image containing the pigeon (second column and third row) a number of separated small regions appears on the pigeon body. In the corresponding ground truth segmentation, those regions are part of the object (the pigeon), however, in our result they are part of the background. This is because the proposed approach is a pixel

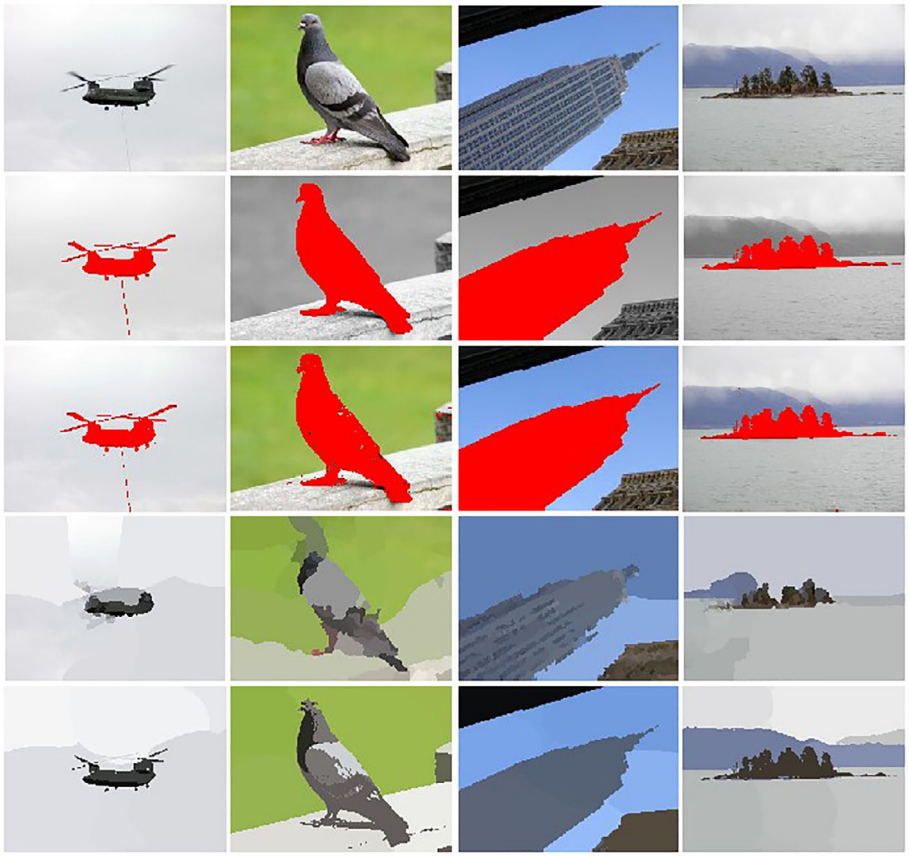

FIGURE 11 Qualitative comparison of the proposed framework segmentation against that of other state-of-the-art techniques using the segmentation evaluation dataset. From upper to lower rows: original images, expert manual segmentations, segmentations using echo state network-based framework, segmentations using normalized cuts,[35] segmentations using mean shift,[33] respectively

classification-based technique, where each pixel is assigned to a class according to their features regardless of the label of their neighboring pixels. If the image is noisy, the number of outlier pixels rises and the performance of the segmentation decreases. Such problem can be addressed through the suppression of small areas at the end of the segmentation process. In addition, image preprocessing such as denoising should help decrease the number of outlier pixels and consequently increase the segmentation performance. Note that our segmentation framework does not involve any pre- or post-processing of the input image. Another solution to prevent this is to segment the image based on the classification of the blocks of pixels instead of single ones, that is, features should be extracted for each block of pixels and the entire block of pixels is then assigned to the same class.

Another advantage of our framework is the small number of extracted features from the images. In fact, for each pixel, we have used only five features (the three color channels in addition to the mean and the SD of each pixel with their neighbors). Furthermore, it is also worth noting that the proposed ESN-based framework is a supervised technique. The technique proposed in Reference 57 is an interactive segmentation technique and all the remaining techniques used in this comparison are unsupervised techniques. A comparison of our proposed framework with other supervised techniques is presented in the following subsection.

6.2.2 | Comparison using retinal images

This section shows the results of comparison of the proposed ESN-based vessels segmentation in retinal images with the following supervised techniques:[22,39-42,44] presented early in Section 2. The ESN reservoir parameters are set as follows: the spectral radius is set to 0.01, the connectivity density is set to 0.2 and the reservoir size is set to 100 nodes. We have plotted the received operating characteristic (ROC) curve as shown in Figure 12 and have reported the corresponding area under the ROC curve (AUC) in Table 6.

Table 6 shows the sensitivity (Se), the specificity (Sp), the accuracy (Acc), and the AUC of the different techniques as reported by their authors. Note that the values of Se and Sp correspond to the optimal threshold which gives the maximum Acc. In general, the proposed framework achieves comparable performance with state-of-the-art techniques. For example, the accuracy value achieved by the proposed framework (0.9470) is slightly higher than those of References 38-41 algorithms, which achieved 0.9416, 0.9452, 0.9461, and 0.9441, respectively. However, it is slightly lower than those of References 42,44 which achieved 0.9474 and 0.9480, respectively. It can be seen that Li et al technique[22] outperformed all the other techniques with an accuracy of 0.9527. Considering the AUC measure, the proposed framework scored a value of 0.9555 which is higher than[38,39] techniques which achieved 0.9294 and 0.9520, respectively, and lower than[22,40-42,44] techniques which achieved 0.9648, 0.9747, 0.9738, 0.9588,

FIGURE 12 The received operating characteristic curve of the proposed framework for the digital retinal images for vessel extraction dataset

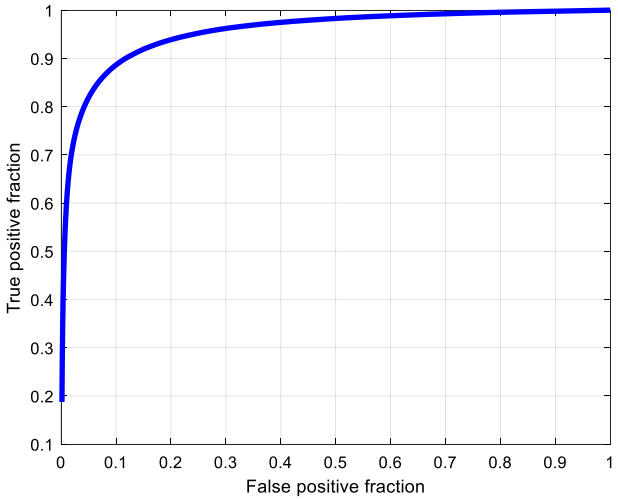

TABLE 6 Comparison of the ESN-based framework segmentation against other state-of-the-art techniques using the DRIVE dataset

Technique	Se	Sp	Acc	AUC
Li[22]	**0.7569**	**0.9816**	**0.9527**	0.9738
Fraz[42]	0.7406	0.9807	0.9480	**0.9747**
Cheng[44]	0.7252	0.9798	0.9474	0.9648
The proposed framework (ESN)	**0.7158**	**0.9791**	**0.9470**	**0.9555**
Soares[40]	0.7332	0.9782	0.9461	0.9614
Marin[41]	0.7067	0.9801	0.9452	0.9588
Staal[39]	NA	NA	0.9441	0.9520
Niemeijer[38]	NA	NA	0.9416	0.9294

Abbreviations: DRIVE, digital retinal images for vessel extraction; ESN, echo state network.

and 0.9614, respectively. The differences between the accuracy of our technique and those of other reported techniques sorted from the best to the worst are −0.0570 (−0.57%), −0.001 (−0.1%), −0.0004 (−0.04%), 0.0009 (0.09%), 0.0018 (0.18%), 0.0029 (0.29%), and 0.0054 (0.54%). The negative values are corresponding to the techniques which give better accuracies than the accuracy of the proposed technique and vice versa. The differences between the accuracy of our technique and those of other techniques lie within a tight range. Therefore, using the proposed framework we have obtained comparable segmentation results with other state-of-the-art techniques.

In addition, the simplicity of the used features by the proposed technique should be highlighted. Table 7 shows the features used by different techniques of blood vessel segmentation in retinal images. The dimension of feature space used by our proposed technique is lower comparing with the other techniques. In fact, we have used only seven low level features (the three chromatic channels, the mean, the SD, and the gradient filter represented by its magnitude and direction) as we discussed in Section 5.2. In Reference 44 Cheng et al have used a large pool of features containing more than 50 components. It includes heterogeneous context-aware features represented by the SWT and WLD in addition to classical local features such as intensity-based features, vesselness, and Gabor-based features. Niemeijer et al[38] used a 31-components feature vector that consists of the Gaussian and its first and second derivatives at five different scales. Staal et al[39] proposed a ridge-based vessel detection approach which computes 27 features for each pixel. In Reference 42, Fraz et al have used a feature space with a dimension of nine components only. It includes gradient-based features, morphological features, line-based features, and Gabor filter-based features. In addition, the dimension of features used by Soares et al[40] is low. They have used a green channel component and Gabor filter responses. However, the computation of the responses of Gabor filter for different scales and orientations is time consuming which increases the computational complexity of these techniques. Therefore, achieving comparable results with other state-of-the-art techniques while using rather simpler features confirms once again the good quality of the final features extracted by the ESN reservoir.

Moreover, samples of the resulting segmentations by the proposed ESN-based approach are shown in Figure 13. The figure shows that the segmentations obtained by the proposed framework are close to the ground truths. However, the segmented images show that our proposed technique reveals a weakness in the detection of thin vessels. In addition, a problem of over segmentation is found around the optic disk area, which appears clearly in the segmentations of the first and the fourth images of Figure 13. A separate segmentation of the optic disk can help remedy this problem.

TABLE 7 Image features used by different techniques for blood vessel segmentation in retinal images

Technique	Features	Features Vector Dimension
Cheng[44]	Stroke width transform, Weber's local descriptors, pixel intensity, Vesselness, and Gabor responses	More than 50
Niemeijer[38]	Green channel component, Gaussian filter and its derivatives up to order 2 at different scales	31
Staal[39]	Ridge-based vessel detection	27
Fraz[42]	Gradient vector field, morphological transformation, line feature, and Gabor responses	9
Our framework	HSV color space components, mean*, SD*, gradient magnitude, and direction	7
Soares[40]	Green channel component and Gabor responses	N.A
Marin[41]	Gray level features (pixel intensity, min of intensities*, max of intensities*, mean*, SD*, first and second moment invariant	7

Note: Features mentioned by (*) are computed between each pixel and its neighbours within a given window.

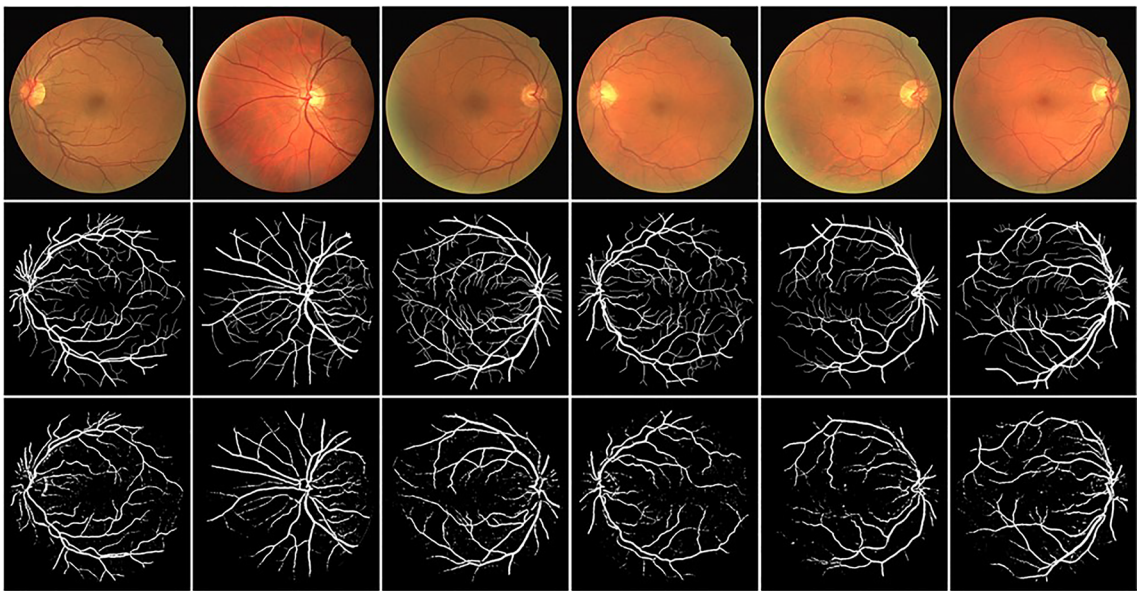

FIGURE 13 Segmentation results of the proposed ESN-based framework on the DRIVE dataset. Upper row: original image. Middle row: expert manual segmentation. Lower row: the proposed ESN-based framework segmentation. DRIVE, digital retinal images for vessel extraction; ESN, echo state network

TABLE 8 Average processing time for segmenting one retinal image by different segmentation techniques

Technique	Processing Time
Li[22]	1.2 minutes
Fraz[42]	2 minutes
Soares[40]	3 minutes
Marin[41]	1.5 minutes
Staal[39]	15 minutes
Cheng[44]	Less than 1 minute
The proposed framework (ESN)	**8 seconds**

The percentage of samples used to train our framework is only 2.51% of the training set pixels. However, in Li[22] and Soares[40] techniques 30% and over 20% of pixels have been used for training, respectively. Marin et al[41] used only 0.65% of the training set. They have produced their own training set by hand that is, they have carefully selected the training samples in such a way that all possible patterns (vessel, background, noise) are covered. This strategy is time consuming, requires experience and skills and might need to be repeated should further training using new images be necessary. The average time required to train our framework is approximately 10 minutes. It has been implemented using MATLAB 2015a on an Intel i5-6400 CPU running at 2.7 GHz with 8 GB of RAM. In comparison, the GMM classifier used by Soares[40] and the deep neural network used by Li[22] are trained in approximately 8 and 7 hours, respectively.

Processing time is an important criterion in the assessing of the performance of any image or video processing techniques.[60-62] Table 8 shows the average time required by different techniques to segment one retinal image. All the processing times of the other cited techniques are taken from Reference 22. The processing time of our framework for an input image requires approximately 8 seconds. It consists of the time required to perform the following three steps: the extraction of initial features (which takes about 1.5 seconds), the collection of the ESN reservoir outputs (which takes about 6 seconds) and the classification using the already trained MLP (which takes about 0.5 seconds). The approach is implemented using MATLAB 2015a on an Intel i5-6400 CPU running at 2.7 GHz with 8 GB of RAM. A further efficient implementation could speed up the proposed technique. It appears clearly from Table 8 that our framework is much faster than all other techniques. This is partly attributed to the absence of any pre or post processing of images in our method. Most of the cited techniques require pre- and post-processing times. For instance, in order to eliminate the high contrast difference between the area of FOV and the area outside the aperture, Soares et al[40] have proposed a technique that grows iteratively the region of interest (ROI), which is initially defined by the aperture of the camera. At first, they locate the pixels that are directly adjacent to the ROI. Then, a new intensity value of each of these pixels is obtained which is the mean of intensities of its neighboring pixels within the initial ROI. Consequently, the initial ROI is growing by adding these modified pixels. And the same process continues until a ROI with the desired size is obtained. Marin et al[41] have used three preprocessing operations: removal of the reflex of the vessel central light, homogenization of the image background, and enhancement of the segmented vessel areas. These kinds of processing, in addition to being time consuming, can change the vessel structure and especially risk removing thin vessels. Another key factor contributing to the efficiency of our proposed framework is the use of simple low-level pixel features as shown by Table 7 and discussed early in this section. Regarding the speed, the simplicity and the fact of achieving comparable results comparing with state-of-the-art, our proposed automatic blood vessel segmentation technique could be a good candidate to be used in complete systems of ophthalmic clinical applications.

7 | CONCLUSION

In this work, we have proposed and evaluated an ESN-based framework for color image segmentation. Low level simple features have been extracted from the input images then the ability of the ESN reservoir dynamics to produce novel pixel features suitable for color image segmentation was investigated. A series of experiments were conducted on several real world image datasets to examine the viability of the proposed approach and thoroughly assess the influence of different ESN parameters on the performance of the segmentation. As a result, the optimal operating ranges of the ESN parameters were identified. It was found that a reservoir with a small density of connectivity between neurons (0.2), a small spectral radius (less than or equal to 0.2) and a small reservoir dimension (50-100 neurons) can result in good quality pixel features and help obtain competitive segmentation performance in comparison with state-of-the-art image segmentation techniques.

The current study can also be considered as a practical guideline for adequate tuning of the ESN parameters for future works on color image segmentation. Extensive experiments were also conducted on a domain-specific real world dataset which consists of segmentation of blood vessels in retinal images to further validate the proposed framework and the identified operating ranges of the ESN parameters. The use of this domain-specific

dataset has proved the competitiveness of the proposed ESN-based framework for color image segmentation in terms of segmentation quality and computation speed. Another interesting finding of this work is the ability of a small subset of arbitrarily chosen neurons from the ESN reservoir to produce good quality pixels features which result in accurate segmentation.

Future work aims to investigate the use of LSM which is based on more biologically plausible spiking neuron models instead of the rate-based counterparts used in ESN. Another interesting future direction would be to investigate the use of deep ESNs for feature extraction and their potential in further enhancing the segmentation quality.

ORCID

Abdelkerim Souahlia https://orcid.org/0000-0002-3393-1608
Elhadj Benkhelifa https://orcid.org/0000-0001-6168-2664

REFERENCES

1. Zhu H, Meng F, Cai J, Lu S. Beyond pixels: a comprehensive survey from bottom-up to semantic image segmentation and cosegmentation. *J Vis Commun Image Represent*. 2016;34:12-27. https://doi.org/10.1016/j.jvcir.2015.10.012.
2. Garcia-Lamont F, Cervantes J. Asdrubal Lopez LR. segmentation of images by color features: a survey. *Neurocomputing*. 2018;292:1-27. https://doi.org/10.1016/j.neucom.2018.01.091.
3. Jaglan P, Dass R, Duhan M. A comparative analysis of various image segmentation techniques. Paper presented at: Proceedings of the 2nd International Conference on Communication, Computing and Networking; 2018:359-374; Chandigarh, India: Springer Singapore. doi:https://doi.org/10.1007/978-981-13-1217-5
4. Chouhan SS, Kaul A, Singh UP. Soft computing approaches for image segmentation: a survey. *Multimed Tools Appl*. 2018;77(21):28483-28537. https://doi.org/10.1007/s11042-018-6005-6.
5. Lukoševičius M, Jaeger H. Reservoir computing approaches to recurrent neural network training. *Comput Sci Rev*. 2009;3(3):127-149. https://doi.org/10.1016/j.cosrev.2009.03.005.
6. Verstraeten D, Schrauwen B, D'Haene M, Stroobandt D. An experimental unification of reservoir computing methods. *Neural Netw*. 2007;20(3):391-403. https://doi.org/10.1016/j.neunet.2007.04.003.
7. Jaeger H. The " echo state " approach to analysing and training recurrent neural networks – with an Erratum note. Ger Natl Res Cent Inf Technol GMD Rep 148, 2001:1–47, article-id:9635932.
8. Maass W, Natschläger T, Markram H. Real-time computing without stable states: a new framework for neural computation based on perturbations. *Neural Comput*. 2002;14(11):2531-2560. https://doi.org/10.1162/089976602760407955.
9. Han M, Xu M. Laplacian echo state network for multivariate time series prediction. *IEEE Trans Neural Netw Learn Syst*. 2018;29(1):238-244. https://doi.org/10.1109/TNNLS.2016.2574963.
10. Ramamurthy R, Bauckhage C, Buza K, Wrobel S. Using echo state networks for cryptography. Paper presented at: Proceedings of the International Conference on Artificial Neural Networks; 2017:663-671; Springer, Cham. doi:https://doi.org/10.1007/978-3-319-68612-775
11. Badoni M, Singh B, Singh A. Implementation of echo-state network-based control for power quality improvement. *IEEE Trans Ind Electron*. 2017;64(7):5576-5584. https://doi.org/10.1109/TIE.2017.2677359.
12. Koprinkova-Hristova P, Angelova D, Borisova D, Jelev G. Clustering of spectral images using echo state networks. Paper presented at: Proceedings of the 2013 IEEE International Symposium on Innovations in Intelligent Systems and Applications IEEE INISTA; vol. 2013, 2013:1-5. doi:https://doi.org/10.1109/INISTA.2013.6577633
13. Meftah B, Lézoray O, Benyettou A. Novel approach using echo state networks for microscopic cellular image segmentation. *Cogn Comput*. 2015;8(2):237-245. https://doi.org/10.1007/s12559-015-9354-8.
14. Lukoševičius M. A practical guide to applying echo state networks. *Neural Networks: Tricks of the Trade*. Berlin, Heidelberg / Germany: Springer; 2012:659-686.
15. Caluwaerts K, Wyffels F, Dieleman S, Schrauwen B. The spectral radius remains a valid indicator of the echo state property for large reservoirs. Paper presented at: Proceedings of the 2013 International Joint Conference Neural Networks; 2013:1-6; Dallas, TX. doi:https://doi.org/10.1109/IJCNN.2013.6706899
16. Yildiz IB, Jaeger H, Kiebel SJ. Re-visiting the echo state property. *Neural Netw*. 2012;35:1-9. https://doi.org/10.1016/j.neunet.2012.07.005.
17. Venayagamoorthy GK, Shishir B. Effects of spectral radius and settling time in the performance of echo state networks. *Neural Netw*. 2009;22(7):861-863. https://doi.org/10.1016/j.neunet.2009.03.021.
18. Zhang A, Zhu WEI, Li J. Spiking echo state convolutional neural network for robust time series classification. *IEEE Access*. 2018;7(1):4927-4935. https://doi.org/10.1109/ACCESS.2018.2887354.
19. Mastoi Q, Wah TY, Raj RG. Reservoir computing based echo state networks for ventricular heart beat classification. *Appl Sci*. 2019;9(4):1-17. https://doi.org/10.3390/app9040702.
20. Cuili Y, Junfei Q, Honggui H, Lei W. Design of polynomial echo state networks for time series prediction. *Neurocomputing*. 2018;290:148-160. https://doi.org/10.1016/j.neucom.2018.02.036.
21. Decai L, Min H, Jun W. Chaotic time series prediction based on a novel robust echo state network. *IEEE Trans Neural Netw Learn Syst*. 2012;23(5):787-799. https://doi.org/10.1109/TNNLS.2012.2188414.
22. Li Q, Feng B, Xie L, Liang P, Zhang H, Wang T. A cross-modality learning approach for vessel segmentation in retinal images. *IEEE Trans Med Imaging*. 2016;35(1):109-118. https://doi.org/10.1109/TMI.2015.2457891.
23. Bozhkov L, Koprinkova-hristova P, Georgieva P. Learning to decode human emotions with echo state networks. *Neural Netw*. 2016;78:112-119. https://doi.org/10.1016/j.neunet.2015.07.005.
24. Sun L, Jin B, Yang H, Tong J, Liu C, Xiong H. Unsupervised EEG feature extraction based on echo state network. *Inf Sci*. 2018;475:1-17. https://doi.org/10.1016/j.ins.2018.09.057.

25. Chouikhi N, Ammar B, Alimi AM. Genesis of Basic and Multi-Layer Echo State Network Recurrent Autoencoder for Efficient Data Representations; 2018:1-13. arXiv Prepr arXiv:1804.08996.
26. Otsu N. A TIreshold selection method from gray-level histograms. *IEEE Trans Syst Man Cybern*. 1979;9(1):62-66. https://doi.org/10.1109/TSMC.1979.4310076.
27. Sarkar S, Das S, Chaudhuri SS. Multi-level thresholding with a decomposition-based multi-objective evolutionary algorithm for segmenting natural and medical images. *Appl Soft Comput*. 2017;50:142-157. https://doi.org/10.1016/j.asoc.2016.10.032.
28. Adams R, Bischof L. Seeded region growing. *IEEE Trans Pattern Anal Mach Intell*. 1994;16(6):641-647. https://doi.org/10.1109/34.295913.
29. Rahini KK, Sudha SS. Review of image segmentation techniques: a survey. *Int J Adv Res Comput Sci Softw Eng*. 2014;4(7):842-845.
30. Senthilkumaran N, Rajesh R. Edge detection method for image segmentation – a survey of soft computing approaches. *Int J Recent Trends Eng Technol*. 2009;1(2):250-254.
31. Li H, He H, Wen Y. Dynamic particle swarm optimization and K-means clustering algorithm for image segmentation. *Opt J Light Electron Opt*. 2015;126(24):4817-4822. https://doi.org/10.1016/j.ijleo.2015.09.127.
32. Liu G, Zhang Y, Wang A. Incorporating adaptive local information into fuzzy clustering for image segmentation. *IEEE Trans Image Process*. 2015;24(11):3990-4000. https://doi.org/10.1109/TIP.2015.2456505.
33. Comaniciu D, Meer P. Mean shift: a robust approach toward feature space analysis. *IEEE Trans Pattern Anal Mach Intell*. 2002;24(5):603-619. https://doi.org/10.1109/34.1000236.
34. Wu Z, Leahy R. An optimal graph theoretic approach to data clustering: theory and its application to image segmentation. *IEEE Trans Pattern Anal Mach Intell*. 1993;15(11):1101-1113. https://doi.org/10.1109/34.244673.
35. Shi J, Malik J. Normalized cuts and image segmentation. *IEEE Trans Pattern Anal Mach Intell*. 2000;22(8):888-905. https://doi.org/10.1109/CVPR.1997.609407.
36. Almotiri J, Elleithy K, Elleithy A. Retinal vessels segmentation techniques and algorithms: a survey. *Appl Sci*. 2018;8(2):1-31. https://doi.org/10.3390/app8020155.
37. Fraz MM, Remagnino P, Hoppe A, et al. Blood vessel segmentation methodologies in retinal images - a survey. *Comput Methods Prog Biomed*. 2012;108(1):407-433. https://doi.org/10.1016/j.cmpb.2012.03.009.
38. Niemeijer M, Staal J, van Ginneken B, Loog M, Abramoff MD. Comparative study of retinal vessel segmentation methods on a new publicly available database. *Medical Imaging 2004: Image Processing*. San Diego, CA: SPIE; 2004:5370-5379. https://doi.org/10.1117/12.535349.
39. Staal J, Abràmoff MD, Niemeijer M, Viergever MA, Van Ginneken B. Ridge-based vessel segmentation in color images of the retina. *IEEE Trans Med Imaging*. 2004;23(4):501-509. https://doi.org/10.1109/TMI.2004.825627.
40. Soares JVB, Leandro JJG, Cesar RM, Jelinek HF, Cree MJ. Retinal vessel segmentation using the 2-D Gabor wavelet and supervised classification. *IEEE Trans Med Imaging*. 2006;25(9):1214-1222. https://doi.org/10.1109/TMI.2006.879967.
41. Marín D, Aquino A, Gegúndez-Arias ME, Bravo JM. A new supervised method for blood vessel segmentation in retinal images by using gray-level and moment invariants-based features. *IEEE Trans Med Imaging*. 2011;30(1):146-158. https://doi.org/10.1109/TMI.2010.2064333.
42. Fraz MM, Remagnino P, Hoppe A, et al. An ensemble classification-based approach applied to retinal blood vessel segmentation. *IEEE Trans Biomed Eng*. 2012;59(9):2538-2548. https://doi.org/10.1109/TBME.2012.2205687.
43. Otoum S, Member S, Kantarci B, Member S, Mouftah HT. Mitigating false negative intruder decisions in WSN-based smart grid monitoring. Paper presented at: Proceedings of the 2017 13th International Wireless Communications and Mobile Computing Conference; 2017:153-158. doi:https://doi.org/10.1109/IWCMC.2017.7986278
44. Cheng E, Du L, Wu Y, Zhu YJ, Megalooikonomou V, Ling H. Discriminative vessel segmentation in retinal images by fusing context-aware hybrid features. *Mach Vis Appl*. 2014;25(7):1779-1792. https://doi.org/10.1007/s00138-014-0638-x.
45. Grant HW, Yu W. A survey of deep learning: platforms, applications and emerging research trends. *IEEE Access*. 2018;6:24411-24432.
46. Otoum S, Member S, Kantarci B, et al. On the feasibility of deep learning in sensor network intrusion detection. *IEEE Netw Lett*. 2019;1(2):68-71. https://doi.org/10.1109/LNET.2019.2901792.
47. Krawiec K. Segmenting Retinal Blood Vessels with Deep Neural Networks. *IEEE Trans Med Imaging*. 2016;35(11):2369-2380. https://doi.org/10.1109/TMI.2016.2546227.
48. Tan JH, Acharya UR, Bhandary SV, Chua KC, Sivaprasad S. Segmentation of optic disc, fovea and retinal vasculature using a single convolutional neural network. *J Comput Sci*. 2017;20:70-79. https://doi.org/10.1016/j.jocs.2017.02.006.
49. Souahlia A, Belatreche A, Benyettou A, Curran K. An experimental evaluation of echo state network for colour image segmentation. Paper presented at: Proceedings of the International Joint Conference on Neural Networks; Vol 2016-October 2016:1143-1150; Vancouver, Canada. doi:https://doi.org/10.1109/IJCNN.2016.7727326
50. Souahlia A, Belatreche A, Benyettou A, Curran K. Blood vessel segmentation in retinal images using echo state networks. Paper presented at: Proceedings of the 9th International Conference on Advanced Computational Intelligence (ICACI 2017); 2017:91-98; Doha, Qatar, IEEE. doi:https://doi.org/10.1109/ICACI.2017.7974491
51. Li H, Cai J, Nhat T, Nguyen A, Zheng J. A benchmark for semantic image segmentation. Paper presented at: Proceedings of the International Conference on Multimedia and Expo (ICME); 2013:1-6; IEEE. doi:https://doi.org/10.1109/ICME.2013.6607512
52. Alpert S, Galum M, Basri R. Brandt a. image segmentation by probalistic bottom-up aggregation and Cue integration. *Proc Conf Comput Vis Pattern Recognit*. 2007;34(2):1-8. https://doi.org/10.1109/CVPR.2007.383017.
53. Alpert S, Galun M, Brandt A, Basri R. Image segmentation by probabilistic bottom-up aggregation and cue integration. *IEEE Trans Pattern Anal Mach Intell*. 2012;34(2):315-327. https://doi.org/10.1109/TPAMI.2011.130.
54. Arbeláez P, Maire M, Fowlkes C, Malik J. Contour detection and hierarchical image segmentation. *IEEE Trans Pattern Anal Mach Intell*. 2011;33(5):898-916. https://doi.org/10.1109/TPAMI.2010.161.
55. Phung SL, Bouzerdoum A, Chai D Sr. Skin segmentation using color pixel classification: analysis and comparison. *Pattern Anal Mach Intell IEEE Trans*. 2005;27(1):148-154. https://doi.org/10.1109/TPAMI.2005.17.
56. Aloqaily M, Ridhawi I, Al Salameh HB, Jararweh Y. Data and service management in densely crowded environments: challenges, opportunities, and recent developments. *IEEE Commun Mag*. 2019;57(4):81-87.

57. Bagon S, Boiman O, Irani M. What is a good image segment_ a unified approach to segment extraction, ECCV 2008.pdf. Paper presented at: Proceedings of the *European Conference on Computer Vision*; 2008:30-44;. Springer, Berlin, Heidelberg / Germany.
58. Galun M, Sharon E, Basri R, Brandt A. Texture Segmentation by Multiscale Aggregation of Filter Responses and Shape Elements. Paper presented at: Proceedings of the 9th IEEE International Conference on Computer Vision; 2003:716-723; IEEE. doi:https://doi.org/10.1109/ICCV.2003.1238418
59. Segmentation evaluation database; 2007. http://www.wisdom.weizmann.ac.il/~vision/Seg_Evaluation_DB/.
60. Al-Ayyoub M, Al-Zu'bi S, Jararweh Y, Shehab MA, Gupta BB. Accelerating 3D medical volume segmentation using GPUs. *Multimed Tools Appl*. 2018;77(4):4939-4958. https://doi.org/10.1007/s11042-016-4218-0.
61. Shehab MA, Al-ayyoub M, Jararweh Y. Improving FCM and T2FCM Algorithms Performance using GPUs for Medical Images Segmentation. Paper presented at: Proceedings of the 2015 6th International Conference on Information and Communication Systems (ICICS); 2015:130-135. doi:https://doi.org/10.1109/IACS.2015.7103215
62. Al-hammouri M, Madani B, Aloqaily M, Ridhawi I Al-Jararweh Y. Scalable video streaming for real-time multimedia applications over DDS middleware for future internet architecture. Paper presented at: Proceedings of the 2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA); 2018:1-6; IEEE.

How to cite this article: Souahlia A, Belatreche A, Benyettou A, Ahmed-Foitih Z, Benkhelifa E, Curran K. Echo state network-based feature extraction for efficient color image segmentation. *Concurrency Computat Pract Exper*. 2020;32:e5719. https://doi.org/10.1002/cpe.5719