Mathematical Biosciences and Engineering

MBE, 21(4): 5092–5117.
DOI: 10.3934/mbe.2024225
Received: 06 November 2023
Revised: 16 February 2024
Accepted: 26 February 2024
Published: 04 March 2024

http://www.aimspress.com/journal/MBE

Research article

A novel lightweight deep learning approach for simultaneous optic cup and optic disc segmentation in glaucoma detection

Yantao Song[1,2,*], **Wenjie Zhang**[1,2] **and Yue Zhang**[2]

[1] Institute of Big Data Science and Industry, Shanxi University, Taiyuan 030006, China
[2] School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China

* **Correspondence:** Email: songyantao@sxu.edu.cn; Tel: +8615513866875.

Abstract: Glaucoma is a chronic neurodegenerative disease that can result in irreversible vision loss if not treated in its early stages. The cup-to-disc ratio is a key criterion for glaucoma screening and diagnosis, and it is determined by dividing the area of the optic cup (OC) by that of the optic disc (OD) in fundus images. Consequently, the automatic and accurate segmentation of the OC and OD is a pivotal step in glaucoma detection. In recent years, numerous methods have resulted in great success on this task. However, most existing methods either have unsatisfactory segmentation accuracy or high time costs. In this paper, we propose a lightweight deep-learning architecture for the simultaneous segmentation of the OC and OD, where we have adopted fuzzy learning and a multi-layer perceptron to simplify the learning complexity and improve segmentation accuracy. Experimental results demonstrate the superiority of our proposed method as compared to most state-of-the-art approaches in terms of both training time and segmentation accuracy.

Keywords: fuzzy learning; multi-layer perceptron; neural networks optic disc segmentation; optic cup segmentation; glaucoma screening

1. Introduction

Glaucoma is a chronic neurodegenerative disease that causes irreversible vision loss and significantly impacts one's quality of life [1,2]. According to the World Health Organization, it ranks as the second most prevalent cause of blindness worldwide, trailing only cataracts. What is particularly concerning is the escalating incidence of glaucoma among younger populations. By the year 2020, approximately 76 million individuals had been diagnosed with glaucoma, and this number is projected to surge to 118 million by 2040 [2,3]. However, owing to the inconspicuous nature of early-stage symptoms and the enduring habits of patients, glaucoma often goes undetected until its later stages. Therefore, early screening and diagnosis of glaucoma are crucial for vision preservation. Among these methods, optical coherence tomography (OCT) and retinal fundus imaging stand out as the most widely employed techniques for glaucoma screening. While OCT provides precise data, it is relatively expensive and less accessible than retinal fundus imaging, which is more commonly used for glaucoma detection. Glaucoma affects the retinal fiber layer, leading to alterations in the internal eye structures, which is prominently reflected in an increased optic cup-to-disc (CDR) ratio. The CDR represents the ratio between the size of the optic cup (OC) and the size of the optic disc (OD). A higher CDR can serve as an indicator of potential glaucoma, with a CDR exceeding 0.65 typically warranting suspicion [4]. Consequently, experienced ophthalmologists often rely on CDR assessments for glaucoma screening and clinical evaluations. Figure 1 visually demonstrates the substantial differences in OD and OC sizes between normal eyes and eyes at varying stages of glaucoma. The first column showcases a normal eye, the middle column represents the early stages of glaucoma, and the last column illustrates an advanced stage glaucoma case. Traditionally, ophthalmologists have manually computed the CDR by segmenting the OD and OC in retinal fundus images. Nevertheless, the sheer volume of images generated daily renders manual processing a time-consuming, costly, and subjectivity-prone endeavor. Therefore, there exists a pressing demand for automated and precise segmentation methods for ODs and OCs in glaucoma studies. Such methods can effectively detect subtle image changes over time, thus enabling early diagnosis.

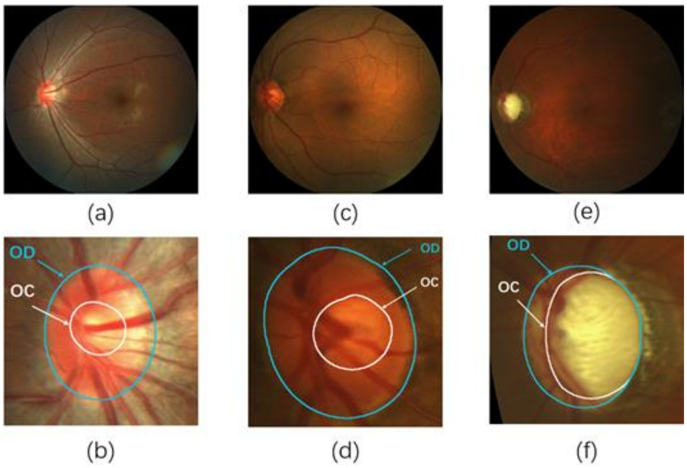

Figure 1. The substantial differences in OD and OC sizes between normal eyes and eyes at varying stages of glaucoma. (a) Image of a normal eye. (b) Boundaries of OD and OC in (a). (c) Early stages of glaucoma. (d) Boundaries of OD and OC in (c). (e) Advanced stages of glaucoma. (f) Boundaries of OD and OC in (e).

Segmentation errors may arise when dealing with bright objects, such as exudates and noise in fundus images, as they often exhibit high intensity values similar to those of OD and OC regions. Additionally, varying illumination conditions can lead to poor contrast and reduced resolution in retinal fundus images, amplifying the complexity of the segmentation process. Furthermore, the presence of blood vessels within the OC region, typically situated within the OD region and referred to as cupping [5], adds another layer of complexity to the process of accurately delineating OC boundaries that does not exist for the OD region.

Over the past two decades, various methods have been developed for OD and OC segmentation [6–10]. These approaches can be broadly categorized into two groups: traditional image processing-based methods and deep learning-based methods. Traditional methods, including thresholding, the level set, active contour modeling, etc., primarily rely on hand-crafted features such as color, texture, contrast, and gradient. Paper [11] and [12] leveraged variational level sets to detect OD and OC contours. Abdel Ghafar and Morris [13] proposed a threshold-based segmentation method after utilizing morphological operations and Lee filters for OD region extraction. Pathan et al. [9] employed a decision tree classifier to realize adaptive thresholding for OD contour detection. Snake-based active contour modeling has been used in [14] to minimize an energy function, gradually converging to object edges. Lalonde et al. [15] adopted a Hausdorff-based template matching technique in combination with multiresolution image decomposition to localize OD regions.

Despite their successes, classical image processing methods are subject to the quality of fundus images and the inherent limitations of hand crafted features, which lack depth information, and thus constrain further advancements. The rapid evolution of deep learning has paved the way for convolutional neural networks (CNNs), offering a promising avenue for automatic feature extraction and emerging as a dominant research direction in medical image processing. CNN-based segmentation methods have demonstrated competitive results when compared to traditional techniques, owing to their ability to learn intricate features from data and adapt to varying imaging modalities. As a result, researchers have increasingly turned to CNNs for glaucoma screening tasks.

In the context of OD and OC segmentation tasks, the authors of [16] introduced a Transformer-based segmentation network that offers an expansive perceptual field, even at high levels of feature resolutions. Complementing this, The SeATrans model developed by Wu et al. [17] represents an asymmetric multiscale network, effectively correlating individual low-level features with multiscale counterparts and demonstrating promising outcomes in OD and OC segmentation. Similarly, multi-layer perceptron(MLP) based networks have garnered much attention for various computer vision applications [18–20]. Among these, the MLP-Mixer proposed by Tolstikhin et al. [20] is the most representative one, achieving comparable performance to the Vision Transformer [21] with fewer parameters. The UNeXt model proposed by Valanarasu and Patel [22] has made a notable contribution to the field. UNeXt introduces a tokenized MLP block to label and project features extracted by the CNN. This incorporation of a MLP enables effective modeling of representation features, leading to competitive performance across a spectrum of tasks. Despite the remarkable progress made by deep learning-based methods in OD and OC segmentation, they still face challenges in the area of meeting the strict requirements for segmentation accuracy in medical images. Furthermore, as segmentation results improve, methods tend to grow in complexity, with an exponential increase in the number of parameters. Consequently, extensive training times are required, even on high-performance computing systems equipped with ample GPU resources are used, and this is an essential consideration for real-time applications.

Fuzzy learning has emerged as a powerful tool to address feature ambiguity in data understanding and classification tasks, as evidenced by its application in various studies [23–26]. For example, Zhou et al. [24] performed a latent space transformation of raw data followed by the fuzzification of deep representations in the output layer for pattern classification. Similarly, the research in [25] and [26] leveraged fuzzy degrees to generate high-level summarizations of input data. Given the effectiveness of both neural networks and fuzzy theory in data representation, the integration of neural learning and fuzzy learning principles is a natural progression. Early attempts in this direction have been explored in studies such as [27–29].

Based on the aforementioned considerations, we present an innovative end-to-end segmentation network tailored for OD and OC segmentation. Our proposed approach leverages fuzzy learning in conjunction with MLPs to extract information from both fuzzy and neural representations. This approach simplifies learning complexity and enhances segmentation accuracy. Notably, our design incorporates two distinct MLP modules, facilitating the capture of non-local features and expanding the perceptual field while concurrently boosting network speed. The contributions of our proposed method are threefold: 1) Our approach seamlessly integrates fuzzy theory with deep learning, providing insights into the effectiveness and limitations of combining fuzzy logic and deep learning. It furnishes evidence of the potential advantages and challenges inherent in this fusion. 2) We introduce and detail two unique MLP modules. These modules excel at capturing fine details in lower layers and extending the perceptual field to encompass depth information in higher layers. This innovation substantially improves segmentation accuracy. 3) Our network structure allows for reduced complexity, resulting in faster inference times than the state-of-the-art methods, all while maintaining a high degree of segmentation accuracy. Moreover, our framework boasts flexibility, allowing seamless integration with various neural network-based segmentation methods for performance enhancement.

The remainder of the paper is organized as follows. In Section 2, we provide a concise introduction to existing fuzzy learning methods and the MLP architecture within the literature. Section 3 presents an in-depth introduction to our proposed network architecture and a detailed description of the design of each module. In Section 4, we present the experimental results, complete with a comprehensive comparative analysis that includes state-of-the-art techniques. Finally, we conclude our work, summarizing our contributions and discussing potential avenues for future research in Section 5.

2. Related works

2.1. Fuzzy learning

The concept of fuzzy sets was first proposed by Zadeh in 1965 [30], marking a significant milestone in the effort to address the inherent imprecision and ambiguity that is pervasive in many real-world problems [31,32]. Building upon the foundational concept of fuzzy sets, fuzzy theory has emerged as a mathematical framework that is capable of effectively managing uncertainties that are inherent in raw data across a spectrum of practical applications. Within the medical domain, for instance, the presence of elevated noise levels in imaging data and the unpredictability stemming from data ambiguity present formidable challenges in medical data processing. Fuzzy theory has been adeptly harnessed to confront these challenges, enhancing the accuracy and resilience of medical data analysis and decision-making processes.

In fuzzy learning-based systems, the representation of features often relies on fuzzy membership functions. These functions encapsulate the degree of membership of data points within the derivation of fuzzy sets. Subsequently, these fuzzy membership functions serve as the foundation for deriving fuzzy reasoning rules that establish the relationship between input features and output decisions. These rules may be expert-defined or learned from data through machine learning techniques. The culmination of these fuzzy reasoning rules takes place in a defuzzifier, which aggregates the fuzzy logic values, ultimately yielding a definitive decision or output. This iterative process gracefully accommodates imprecise and uncertain information in a versatile and interpretable manner. Consequently, fuzzy-based systems are remarkably effective in scenarios in where conventional crisp logic falls short, notably in tasks like medical diagnosis, decision support systems, and pattern recognition.

In contrast to traditional non-fuzzy neurons, which are characterized by multiple inputs and a singular output, fuzzy neurons embedded within fuzzy-based systems forge a nuanced connection between each output and the membership value of a fuzzy concept. This membership value represents the degree to which an input pattern belongs to a particular fuzzy set. The fuzzy neuron computes the weighted sum of its inputs by using corresponding weights, and it passes the result through an activation function. The weights, activation thresholds, and output functions of fuzzy neurons collaboratively describe the interactions between them, and they can be adjusted during the learning process to adapt and improve the performance of the system. This adaptability allows fuzzy-based systems to effectively handle imprecise and uncertain information, making them suitable for a wide range of applications in various fields, including medical data processing, pattern recognition, and decision support systems [23].

2.2. MLP architecture

Artificial neural networks (ANNs) represent computational models that are designed to emulate the learning mechanisms observed in biological neural systems. They were initially proposed based on the concept of the perceptron model, and they consist of interconnected neurons that exhibit similar characteristics to the human brain's neural systems. ANNs can be trained and learn from experience, allowing them to compute complex relationships between neurons and process nonlinear, massively distributed information. A feedforward ANN with supervised learning can adaptively approximate any nonlinear mapping function, making it capable of modeling intricate relationships between inputs and outputs [33]. As a result, they have demonstrated significant success in solving image processing problems due to their superior learning and generalization capabilities [34–38].

Among the various types of neural networks, the feed forward MLP reigns as a widely used and adaptable model. A quintessential MLP architecture comprises an input layer, one or more hidden layers, and an output layer. The pivotal strength of an MLP lies in its aptitude for generating nonlinear function mappings through both its hidden and output layers. This attribute positions the MLP ideally for handling intricate network structures and achieving exceptional accuracy with relatively modest training iterations. In the MLP neural network paradigm, each unit computes a weighted sum of its inputs, subsequently subjecting the result to a nonlinear activation function en route to the output layer. The iterative training process hinges on the fine-tuning of weights via backpropagation, effectively minimizing prediction errors. This cyclical interplay continues until the prediction error converges to a stable value.

Many theoretical and experimental studies have consistently demonstrated that a single hidden layer suffices for the approximation of complex nonlinear functions in MLP models [39,40].

Consequently, many studies have gravitated toward a single hidden layer's simplified yet highly effective structure. Figure 2 illustrates the architecture of a typical three-layer MLP model.

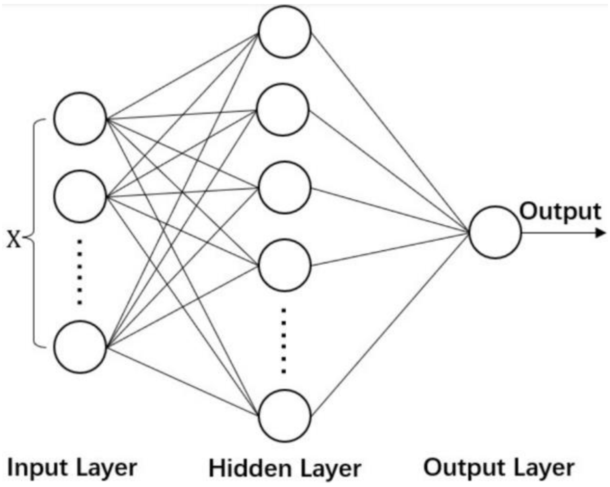

Figure 2. Three-layer MLP architecture.

The mathematical function of MLP with one hidden layer can be formulated as:

$$g(x) = w_2 f(w_1 x + b_1) + b_2 \tag{1}$$

where X denotes the input variable, f is the activation function, and w_1 and b_1 correspond to the weight and bias matrices between the input and the hidden layer, respectively. Similarly, w_2 and b_2 are the weight and bias matrices between the hidden and the output layer, respectively. From Eq (1), it can be seen that the hidden units in the MLP play an important role in the determination of the output values. This inherent characteristic of MLPs affords them the ability to obtain higher accuracy than linear classifiers.

3. Proposed method

We proposed an innovative asymmetrical downsampling-upsampling network structure, as illustrated in Figure 3. Taking an RGB image reshaped to 256 × 256 as an example, the first downsampling layer results in a feature map with dimensions of 128 × 128 and a channel count of 32. The initial downsampling includes a fuzzy module and an SE-Block. The primary focus of the fuzzy module is to capture high-quality fuzzy features, while the SE-Block is a well-known lightweight channel attention module that has shown significant performance gains in previous work such as MobileNetV3 [41] and PP-LCNet [42].

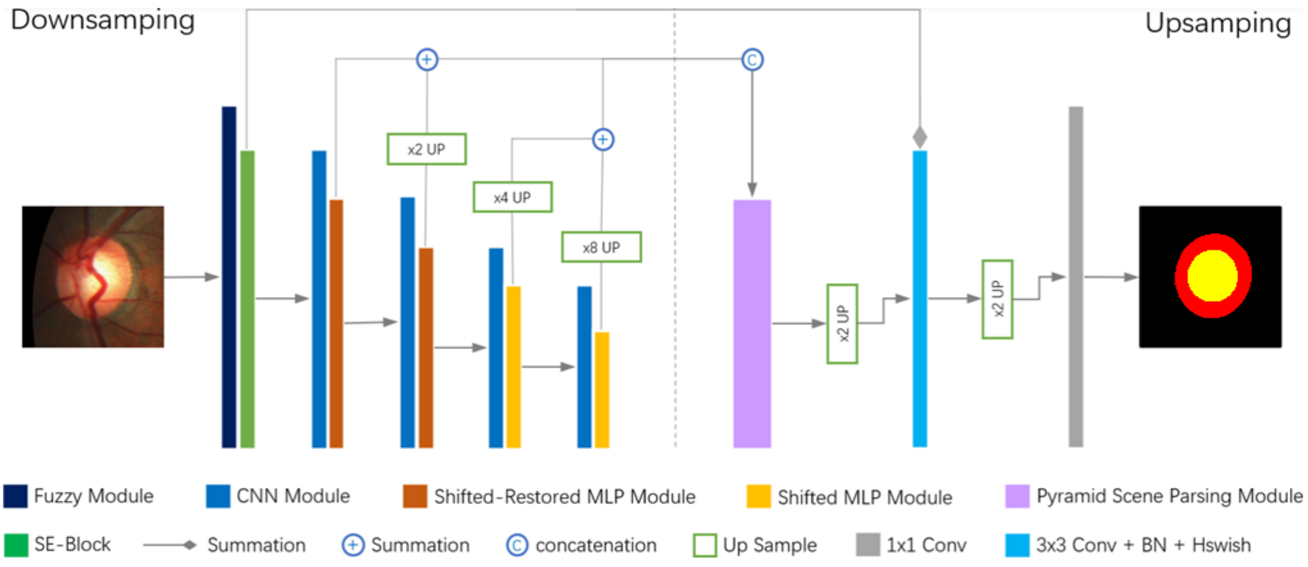

Figure 3. Flowchart of the proposed algorithm.

In the subsequent downsampling from the second to the fifth layer, we employ a CNN+MLP structure to simplify the complexity of the network while ensuring accuracy. A compact CNN convolutional operations is used as the backbone network for feature extraction, and only a limited number of features are extracted (The resolution of the feature maps output after each downsampling layer is modified to be 64, 64, 96, 96, respectively. Additionally, the image resolution after each downsampling layer is reduced by 50%.) Although the MLP is inferior to the CNN in terms of feature extraction, it has fewer parameters, faster inference, and is ideal for labeling features to improve feature quality. Within this framework, we have designed two distinct MLP modules: shifted MLP(S-MLP) and shifted-restored MLP (SR-MLP), SR-MLP exhibits greater sensitivity to location information, rendering it advantageous for processing lower-layer details and contours. In contrast, S-MLP can encompass a broader perceptual field, making it better suited for capturing depth information at higher layers.

During the upsampling process, to comprehensively integrate feature maps of different scales, the feature maps output from the third to the fifth layers are upsampled to the same size as the second layer, i.e., 64 × 64. Subsequently, the feature maps with the same channel counts (64 for the 2nd and 3rd layers, and 96 for the 4th and 5th layers) are added element-wise to create two new feature maps with identical sizes. These two feature maps are then stacked and input into the PSP module, which fuses hierarchical information and outputs a feature map with a resolution of 64 × 64 and a channel count of 96. After an additional upsampling to a resolution of 128 × 128 with 96 channels, a convolutional module is applied to reduce the channel count from 96 to 32. Finally, the feature map is upsampled to the input size and a 1 × 1 convolution is employed to generate the predicted image. Further elaboration on the procedures within each module is provided in the subsequent sections.

3.1. Fuzzy module

The introduced fuzzy module initiates the process by generating feature maps via convolution with a stride of 2. Each node within the input layer of the fuzzy module establishes connections with multiple fuzzy membership functions. These functions are instrumental in computing the degree of fuzziness associated with each input node associated with a specific fuzzy set. The quantification of fuzziness for each feature map is determined through the employment of Gaussian functions. The formula is given as follows,

$$o_i = e^{-(f_{Conv}(x_i)-u_i)^2/\sigma_i^2} \tag{2}$$

where, x_i is the i-th node of the input, f_{Conv} refers to the convolution operation of the first layer, and u_i and σ_i^2 denote the mean and variance of the i-th feature map, respectively. o_i is the output fuzziness map of the i-th feature map. The overall structure of the fuzzy module is shown in Figure 4.

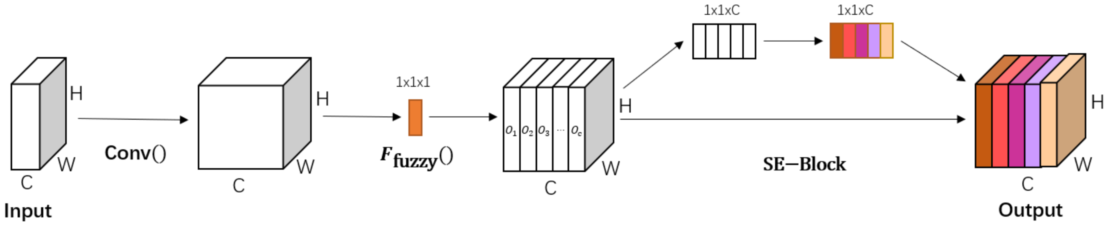

Figure 4. The structure of the fuzzy module.

From another perspective, the fuzziness map can be interpreted as a higher-quality feature map, which can effectively reduce the uncertainty of the input data. However, it is worth noting that as the number of network layers increases, the effectiveness of fuzzy learning decreases significantly. On the other hand, although fuzzy learning requires only a few additional parameters, it demands substantial computation. Consequently, in order to balance the accuracy and speed, fuzzy learning is only applied in the first layer of downsampling within our proposed model.

3.2. CNN module

Our convolution module comprises a single 3 × 3 convolution, accompanied by batch normalization, maximum pooling, and the H-Swish activation function [41], as depicted in Figure 5. Alternatively traditional methods like ResNet [43] have employed two small-sized convolutions to approximate large-sized convolutions, which incurs a notable computational cost. To alleviate this concern, we employ a single small-sized convolution in the CNN module of our model. To mitigate the potential accuracy loss from this reduction, we have incorporated a lightweight MLP into our model, as elaborated in Section 3.3.

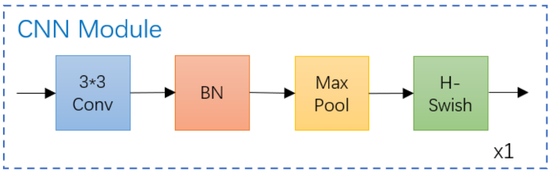

Figure 5. The structure of CNN module.

Additionally, drawing inspiration from the work in [41], we have replaced the ReLU function in our CNN module's activation layer with H-Swish. H-Swish shares similar properties with Swish [44], including the absence of an upper bound, a lower bound, smoothness, and non-monotonicity. However, H-Swish offers faster computational performance than Swish, rendering it advantageous for deep models. The formula for H-Swish is presented below.

$$h - swish[x] = x\frac{ReLU6(x+3)}{6} \tag{3}$$

3.3. MLP module

MLP neural networks are known for their parallel execution capabilities, which allow them to be trained in fewer iterations and effectively handle complex networks. In this study, two novel MLP modules, namely, the S-MLP and SR-MLP, where designed to capture non-local features and enhance the perceptual field, while also improving network speed.

3.3.1. S-MLP module

Motivated by the concept of the moving pane in the Swin Transformer [45], we have developed the S-MLP module to adeptly capture non-local features. The 'S' in S-MLP represents shifted, as the feature maps extracted by the CNN module are divided into five groups and utilized as input. Each set of feature maps undergoes shifting from various directions and scales, accompanied by feature labeling. Figure 6 provides a flowchart for the S-MLP module, using one of the sets of feature maps as an example. As shown in Figure 6, the S-MLP module leverages the shifting of feature maps in different directions and scales to augment the number of channels, thereby capturing a broader range of global features and expanding the feature perceptual field.

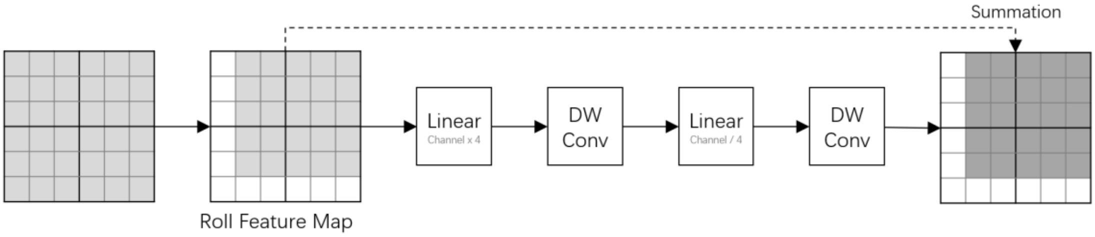

Figure 6. S-MLP module with one of the sets of feature maps.

Subsequent to the shifting operation, the location information of the features is encoded using depth-wise convolution (DWConv), which is recognized for its efficiency in terms of minimal parameters and rapid processing. The structure of DWConv is illustrated in Figure 7. After the DWConv operation, we employ the H-Swish activation function [41] to further enhance the module's performance.

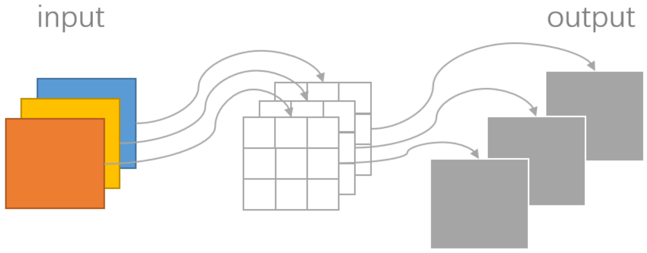

Figure 7. The structure of DWConv.

The mathematical function of the S-MLP block can be summarized as:

$$X_s = Shift_{w,h}(X) \tag{4}$$

$$Y_{S-MLP} = DWConv(MLP\left(DWConv\left(MLP(X_s)\right)\right)) + X_s \tag{5}$$

where *Shift* refers to a shifting operation, *X* is the input, and *h* and *w* done height and width respectively. Y_{S-MLP} is the output feature map.

3.3.2. SR-MLP module

Another MLP module in our architecture is called the SR-MLP, where the capital 'R' in the abbreviation denotes restored. Compared to S-MLP, SR-MLP incorporates an additional restoration step. The flowchart of SR-MLP module with one of the sets of feature maps can be seen in Figure 8.

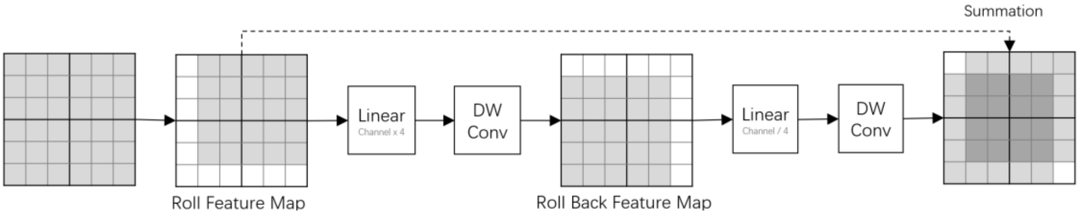

Figure 8. SR-MLP module with one of the sets of feature maps.

After the first DWConv operation in the SR-MLP, the feature map is restored to its original position to recover the number of channels as input. Accordingly, the computed feature maps are connected to the input feature maps by using residuals. This procedure ensures that each set of feature maps contains local feature information after the residual connection, which is sensitive to the location of features. This allows for effective integration of local and non-local features in the module, enhancing its ability to capture both global and local contextual information for improved performance on complex tasks. The mathematical function of the SR-MLP block can be summarized as:

$$Y_{S-MLP} = DWConv(MLP(f\left(Shift_{w,h}\left(DWConv\left(MLP(X_s)\right)\right)\right))) + X_s \tag{6}$$

In summary, the S-MLP module has been designed to obtain a larger feature perceptual field, while SR-MLP module augments the S-MLP with non-locality, making it advantageous for processing fine details. However, due to the time-consuming nature of the shifting operation, the SR-MLP takes more time than the S-MLP. Therefore, here, the SR-MLP is utilized at shallow layers that contain more detailed information, while the S-MLP method is employed in higher layers to capture depth information through larger receptive fields. This strategic use of the S-MLP and SR-MLP modules provides an effective balance between capturing local and global contextual information while managing computational complexity.

3.4. Downsampling and upsampling

The downsampling phase of our proposed network consists of five layers, each extracting a varying number of features: 32, 64, 64, 96, and 96, respectively. These feature quantities are significantly lower than those for mainstream methods, like those in [43]. In the initial layer of the downsampling phase, we incorporate the fuzzy module and SE-Block. The fuzzy module has been designed to yield higher quality fuzzy features, while the SE-Block, serving as a lightweight channel attention mechanism, enhances feature precision. As mentioned earlier, the Fuzzy + MLP module design is employed in all subsequent downsampling layers for consistent feature extraction. In the concluding downsampling layer, the feature maps with the same number of channels are combined into two sets of feature maps in the form of residuals. This fusion of information empowers the network to capture more comprehensive and representative features during the downsampling process, facilitating the extraction of relevant features for subsequent layers.

The PSP module excels at capturing global and local contextual information [46], with its structure depicted in Figure 9. This module employs average pooling with different kernel sizes (1, 3, 5, 7) to acquire average features across various scales. By combining both global and local contextual information, the network benefits from the smaller-scale features that can supplement important information that is otherwise lost in larger-scale features. This integration enhances the generation of a superior global feature map.

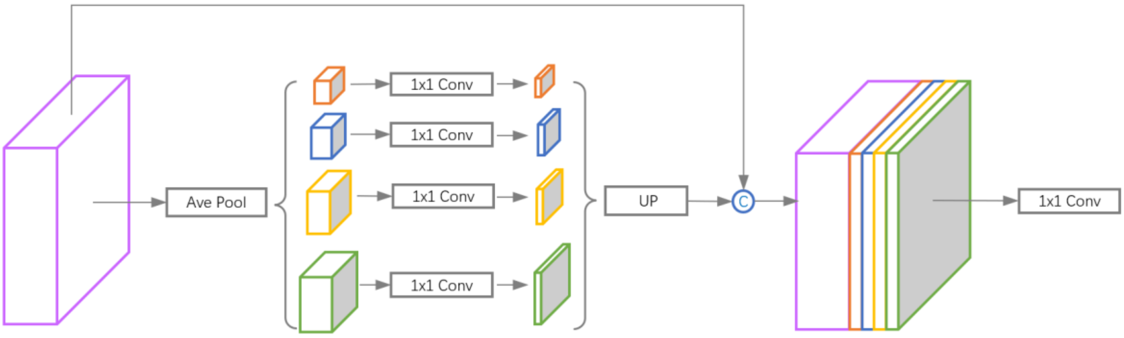

Figure 9. The structure of the PSP module.

To achieve this, the two sets of feature maps merged during the downsampling stage serve as input to the PSP module. While global average pooling effectively extracts contextual information, it can potentially lead to the loss of crucial details. To address this limitation, the PSP module employs average pooling at multiple scales, creating a global + local structure that comprehensively captures both global and local contextual information. This optimized approach ensures a more holistic comprehension of the data by encompassing both global and local features, resulting in enhanced contextual representation.

Despite the significant performance improvements offered by the PSP module, it is computationally intensive. Therefore, for network efficiency, it is essential to minimize feature map sizes whenever possible. In the proposed algorithm, the residual concatenation of feature maps with the same number of channels is performed at the end of downsampling to reduce the computational burden of the PSP module. In the final upsampling layer, upsampling with residual concatenation is employed, and the final prediction is obtained in a 1 × 1 convolutional layer. This approach optimizes the computational efficiency while preserving the performance enhancements achieved by the PSP module.

3.5. Loss function

The selection of loss function is a critical determinant in the performance of segmentation models. In medical applications, false negatives are usually more unacceptable than false positives. In retinal fundus images, the target region (e.g., OD or OC) is typically much smaller than the background region, which leads to an imbalance that may result in more false negatives in the predicted results. To address this issue, the Jaccard index, which measures the overlapping ratio of the foreground mask, can effectively handle the imbalance between foreground and background regions. Therefore, in our proposed model, the Jaccard distance function is incorporated in combination with the cross-entropy function to ensure more accurate segmentation results. Therefore, the loss function of our model is defined as follows, incorporating both the Jaccard distance and cross-entropy terms:

$$L = L_c + kL_j \tag{7}$$

where L_c and L_j denote the weighted cross-entropy and Jaccard losses, respectively. k is the trade-off parameter.

L_c can be further expressed as follows:

$$L_c = -\sum_i W_i Y(i) log_2 P(i) \tag{8}$$

$P(i)$ is the predicted value of classification i. In this study, the weights were set to 0.2, 0.4 and 0.4 for the background, OD and OC respectively.

The loss function for the Jaccard index is as follows:

$$L_j = 1 - \frac{TP}{TP+FP+FN} \tag{9}$$

where TP, FP and FN indicate true positive, false positive and false negative, respectively.

4. Experiments and discussion

4.1. Dataset and metrics

In this study, we conducted experiments on three publicly available datasets, namely DRISGHTI-GS [47], RIM-ONE-R3 [48], and REFUGE [49], to evaluate our proposed method. These datasets provide manual annotations of both the OD and OC as ground truth masks, which were annotated by ophthalmologists or specialists specifically trained for this task. The detailed information of each dataset is presented in Table 1. The DRISGHTI-GS dataset comprises 101 fundus images with a resolution of 2047 × 2056 pixels, while the RIM-ONE-R3 dataset includes 159 fundus images with a resolution of 2144 × 1424 pixels. The REFUGE dataset contains 1200 color fundus images, with 400 images acquired from the Zeiss Visucam 500 fundus camera and 800 images from the Canon CR-2 fundus camera.

Table 1. Datasets information used in the experiment.

Dataset	No. of images	Resolution
DRISHTI-GS	101	2896 × 1944
RIM-ONE-R3	159	2144 × 1424
REFUGE	400	2124 × 2056
REFUGE	800	1634 × 1634

The images in all three datasets were divided into two sets: a training set and a testing set. For the DRISGHTI-GS and REFUGE datasets, we utilized the training and testing sets provided by the respective websites. However, since RIM-ONE-R3 does not provide a pre-defined training set, we randomly divided the dataset into a training set and a testing set with a split ratio of 80 and 20%, respectively. This resulted in a total of 50, 128, and 800 training images for the DRISGHTI-GS, RIM-ONE-R3, and REFUGE datasets, respectively. The remaining images in each dataset were used as the test images for evaluation purposes.

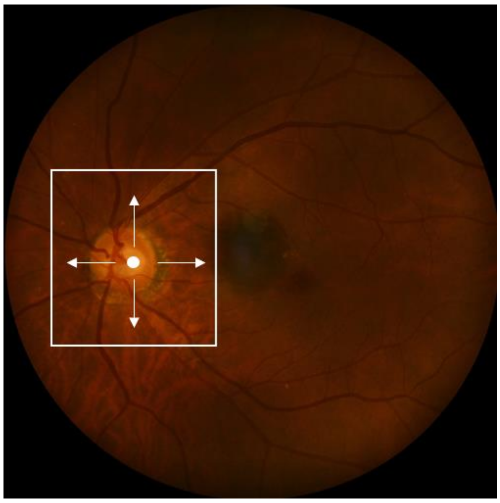

Figure 10. Image cropping method.

To address the issue of limited data in retinal datasets and mitigate overfitting during model training, image cropping and data augmentation techniques were applied to all subjects. First, the images were cropped to a size of 576 × 576 based on their original resolution without extracting the regions of interest for the OD and OC. By analyzing retinal images, it was observed that the OC region typically exhibits the highest brightness. Therefore, utilizing the brightest point in each retinal image as the origin, the images were cropped to a size of 576 × 576, effectively isolating the OD and OC regions, as illustrated in Figure 10. As shown in Table 1, retinal datasets typically have a small number of images; hence, data augmentation was used to augment the training data. Various data augmentation strategies were employed to enrich the original images. Initially, the brightness of the images was randomly adjusted. Subsequently, a random selection was performed with a 30% probability based on six enhancement methods, including horizontal flip, vertical flip, random rotational scaling pan, grid distortion, Gaussian noise, and sharpening. This resulted in expansion of each dataset to 4–10 times the size of the original training set, depending on the volume of the dataset.

To evaluate the performance of the proposed model, five commonly used metrics were employed to evaluate accuracy and stability by comparing the segmentation results with the ground truth. These metrics include the Dice similarity coefficient (DSC), Jaccard coefficient (JC), overall accuracy (OA), and balanced accuracy (BA), which are defined as follows.

The DSC also known as the F1-score, is a measure of the overlap between the predicted segmentation and the ground truth. It is defined as twice the intersection of the predicted and ground truth masks divided by the sum of their areas:

$$DSC = \frac{2TP}{2TP+FP+FN} \tag{10}$$

The JC, also known as the intersection over union, is calculated as the ratio of the intersection of the predicted and ground truth masks to the union of their areas:

$$JC = \frac{TP}{TP+FP+FN} \tag{11}$$

The OA is a statistical measure of the segmentation results, calculated as the ratio of the number of correctly predicted pixels (TP + TN) to the total number of pixels in the image:

$$OA = \frac{TP+TN}{TP+TN+FP+FN} \tag{12}$$

The BA is a measure of accuracy that accounts for imbalanced datasets. It is calculated as the average of the sensitivity (TP rate) and specificity (true negative (TN) rate) of the segmentation results:

$$BA = \frac{1}{2}\left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right) \tag{13}$$

In this case, a positive decision is made when the output is greater than 0.5 otherwise, a negative decision is made. It is worth noting that the DSC and JC are region-based similarity measures used to evaluate segmentation performance, while the OA is a common statistical measure for the same purpose. However, accuracy may not be reliable for imbalanced datasets, which is often the case with imbalanced regions like the OD and OC in retinal datasets. Therefore, in this study, the BA is also utilized as a segmentation performance metric. The BA takes into account both sensitivity and specificity, making it suitable for imbalanced datasets. All of these metrics generate scores between 0 and 1, with higher values indicating better segmentation performance.

4.2. Parameter setting

In the following experiments, the models were trained and tested on the VSCode platform under Ubuntu 18.04 operating system, mainly using the Pytorch framework. We ran our method on

NVIDIA TITAN Xp GPU with 12 GB memory. RMSprop was used to optimize the model during the training process. The learning rate was gradually reduced from 0.0003 with a momentum of 0.9 and a weight decay of 0.000001. The training batch size was set to 4, and 60 epochs were executed for about 45 minutes to achieve convergence.

4.3. Results and comparisons

4.3.1. Quantitative results

We compared our method with Unet [50], pOSAL [51],M-net [52],BGA-NET [53], GDCSeg-Net [54], Fuzzy-BLS [55], CE-Net [56] and Segtran [17] methods in terms of their segmentation of the OD and OC on three datasets: Drishti, rim, and refuge. We evaluated the methods by using four metrics, namely, the DSC, JC, OA, and BA. We re-run the Unet, BGA-NET, GDCSeg-Net, CE-Net, and Segtran methods with the same training and testing set split. As no code was provided for the Fuzzy-BLS method, we used the data from their paper. As a result, we chose to use “N/A: Not Available” to replace the corresponding results that were not provided in their paper.

Tables 2–4 demonstrates that the proposed method outperforms most of the other algorithms on all metrics for OD and OC segmentation. Notably, our method achieved higher accuracy on the task of segmenting the OD than the OC, possibly due to the presence of blood vessels and the low-contrast boundary of the OC region. Importantly, our method outperforms other techniques in accurately segmenting the OC region, highlighting its ability to handle low-quality images and capture intricate details.

Table 2. Comparison of different methods on the drishti dataset (N/A: Not available).

Method	OD				OC			
	DSC	JC	BA	OA	DSC	JC	BA	OA
UNet [50]	0.9408	0.8901	0.9773	0.9704	0.8546	0.7518	0.9058	0.9712
pOSAL [51]	0.9650	N/A	N/A	N/A	0.8580	N/A	N/A	N/A
M-net [52]	0.9678	0.9386	N/A	N/A	0.8618	0.7730	N/A	N/A
BGA-Net [53]	0.9714	0.9448	0.9783	0.9884	0.9081	0.8349	0.9448	0.9849
GDCSeg [54]	0.9711	0.9443	0.9806	0.9886	0.9084	0.8348	0.9339	0.9821
Fuzzy-BLS [55]	0.9680	N/A	N/A	N/A	0.8800	N/A	N/A	N/A
CE-Net [56]	0.9734	0.9486	0.9853	0.9891	0.9087	0.8353	0.9498	0.9857
Segtran [17]	0.9747	0.9508	0.9834	0.9899	0.8921	0.8168	0.9552	0.9842
Proposed	**0.9757**	**0.9528**	**0.9870**	**0.9901**	**0.9202**	**0.8546**	**0.9559**	**0.9872**

Table 3. Comparison of different methods on the rim dataset (N/A: Not available).

Method	OD				OC			
	DSC	JC	BA	OA	DSC	JC	BA	OA
UNet [50]	0.8850	0.7992	0.9648	0.9704	0.7160	0.5760	0.8789	0.9817
pOSAL [51]	0.8650	N/A	N/A	N/A	0.7870	N/A	N/A	N/A
M-net [52]	0.9526	0.9114	N/A	N/A	0.8348	0.7300	N/A	N/A
BGA-Net [53]	0.9699	0.9418	0.9797	0.9884	0.8807	0.7922	0.9263	0.9869
GDCSeg [54]	0.9624	0.9288	0.9775	0.9858	0.8519	0.7528	0.9210	0.9827
Fuzzy-BLS [55]	0.9730	N/A	N/A	N/A	0.8820	N/A	N/A	N/A
CE-Net [56]	0.9735	0.9487	0.9851	0.9897	0.8772	0.7869	0.9300	0.9872
Segtran [17]	0.9736	0.9489	0.9821	0.9898	0.8762	0.7865	0.9242	0.9875
Proposed	**0.9747**	**0.9509**	**0.9858**	**0.9902**	**0.9044**	**0.8286**	**0.9493**	**0.9900**

Table 4. Comparison of different methods on the refuge dataset (N/A: Not available).

Method	OD				OC			
	DSC	JC	BA	OA	DSC	JC	BA	OA
UNet [50]	0.9486	0.9036	0.9751	0.9766	0.8654	0.7676	0.9257	0.9838
pOSAL [51]	0.9460	N/A	N/A	N/A	0.8750	N/A	N/A	N/A
BGA-Net [53]	0.9540	0.9126	0.9876	0.9881	0.8864	0.8002	0.9630	0.9927
GDCSeg [54]	0.9546	0.9139	0.9874	0.9883	0.8929	0.8105	0.9590	0.9931
ET-Net [57]	0.9529	N/A	N/A	N/A	0.8912	N/A	N/A	N/A
Fuzzy-BLS [55]	**0.9743**	N/A	N/A	N/A	0.8845	N/A	N/A	N/A
CE-Net [56]	0.9599	0.9234	**0.9893**	0.9897	0.8959	0.8152	0.9519	0.9933
Segtran [17]	0.9608	0.9251	0.9872	0.9899	0.8974	0.8177	0.9567	0.9933
Proposed	0.9642	**0.9313**	0.9877	**0.9910**	**0.8982**	**0.8189**	**0.9576**	**0.9935**

4.3.2. Visualization results

As shown in Figure 11, we present the qualitative results of our method compared to Unet [50], BGA-NET [53], GDCSeg-Net [54], CE-Net [55], and Segtran [17] on three datasets and the task of OD and OC segmentation in retinal images; we have used white borders to depict the gold standard edge in the segmentation diagram. The first three rows depict normal images without glaucoma from the REFUGE dataset. The middle three rows showcase early-stage glaucoma images from the RIM dataset. The last three rows exhibit advanced-stage glaucoma images from the DRISHTI dataset. The first column shows the original fundus image, and from the second to the sixth column, the segmentation results of the compared methods are displayed. The second to last column shows the result of our proposed method, and the last column shows the ground truth result.

Upon examination of Figure 11, it can be observed that all methods achieved good segmentation results for normal cases. For glaucoma cases, our segmentation results have smoother edges than other methods and are basically consistent with ground truth segmentation result. Therefore, our method provides more reliable segmentation results for both the fundus and orbital regions than the other techniques. Consistent with the above conclusion, due to the intrinsic characteristics of retinal images, the segmentation accuracy for the OD is slightly lower than that for the OC.

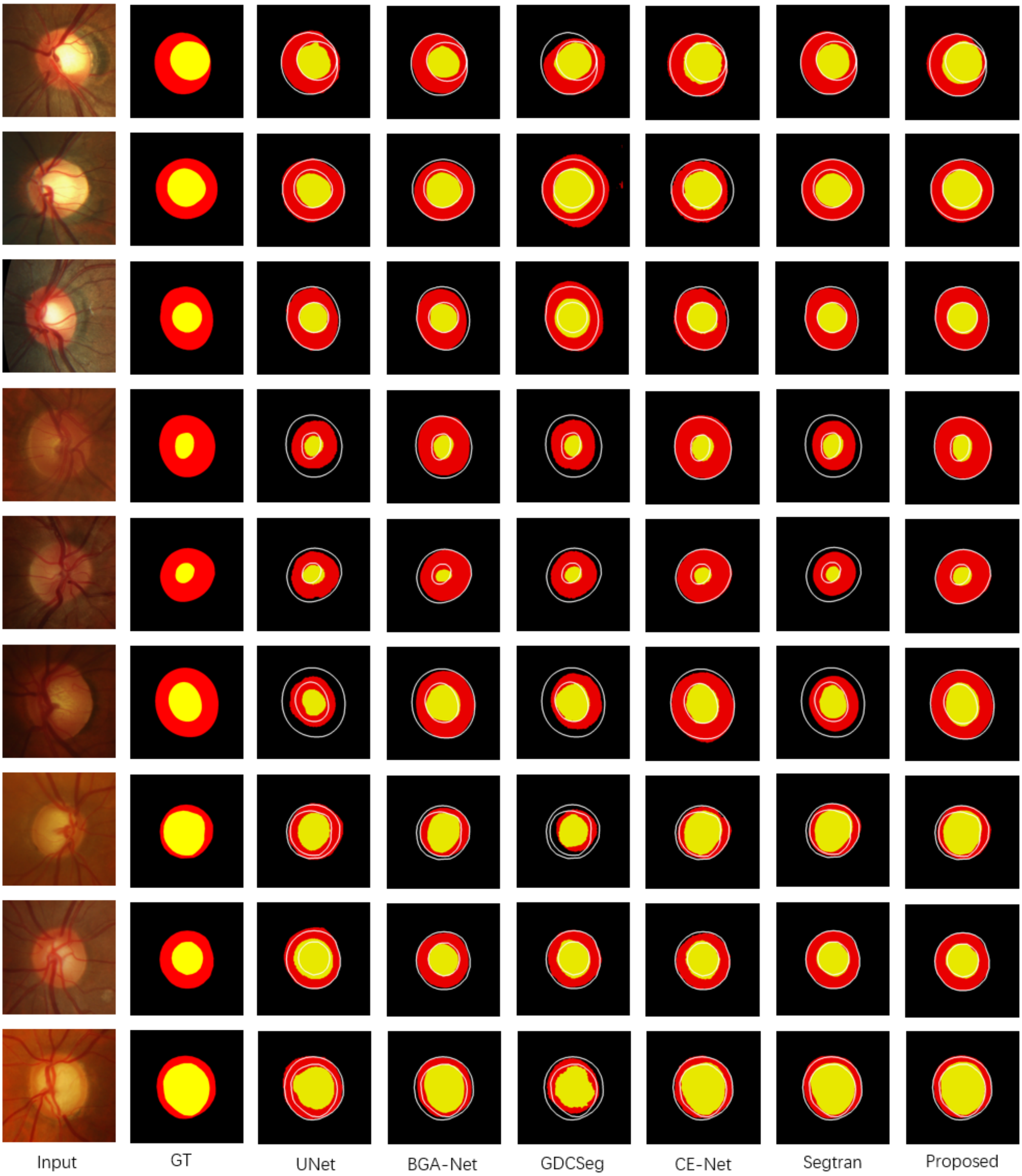

Figure 11. The visual examples of the OD and OC segmentation results, where the yellow and red regions denote the OC and OD segmentation, respectively.

In Figures 12–14, we present the loss curves that delineate the training and testing processes of our proposed method across three distinct datasets: REFUGE, RIM, and DRISHTI. Each figure corresponds to a specific dataset and effectively illustrates the convergence trends that characterize the model's optimization performance. The observed patterns in the curves indicate the model's capacity to learn and adapt to the inherent characteristics of each dataset.

Notably, the REFUGE dataset, distinguished by its significantly larger volume of data compared to RIM and DRISHTI, resulted in an accelerated convergence on the validation set. This expedited convergence on the REFUGE dataset is not indicative of an increased speed of convergence rather, it is a consequence of its extensive data size. Despite the fact that each epoch processes a larger quantity of data, the convergence speed remained consistent. The sheer abundance of data in the REFUGE dataset facilitates a more comprehensive learning process, allowing the model to achieve comparable performance to RIM's 10 epochs in just one epoch.

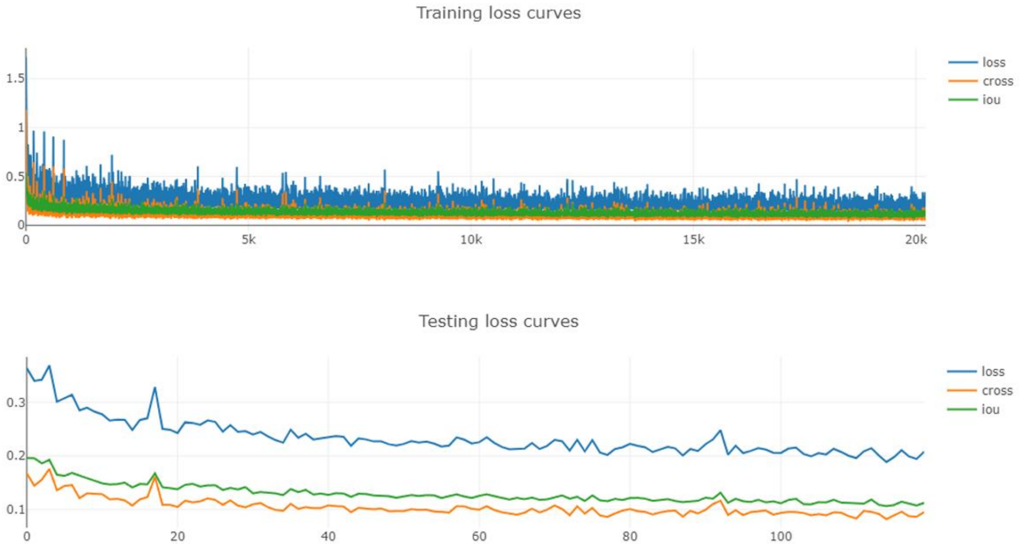

Figure 12. The loss curves of training and testing on refuge.

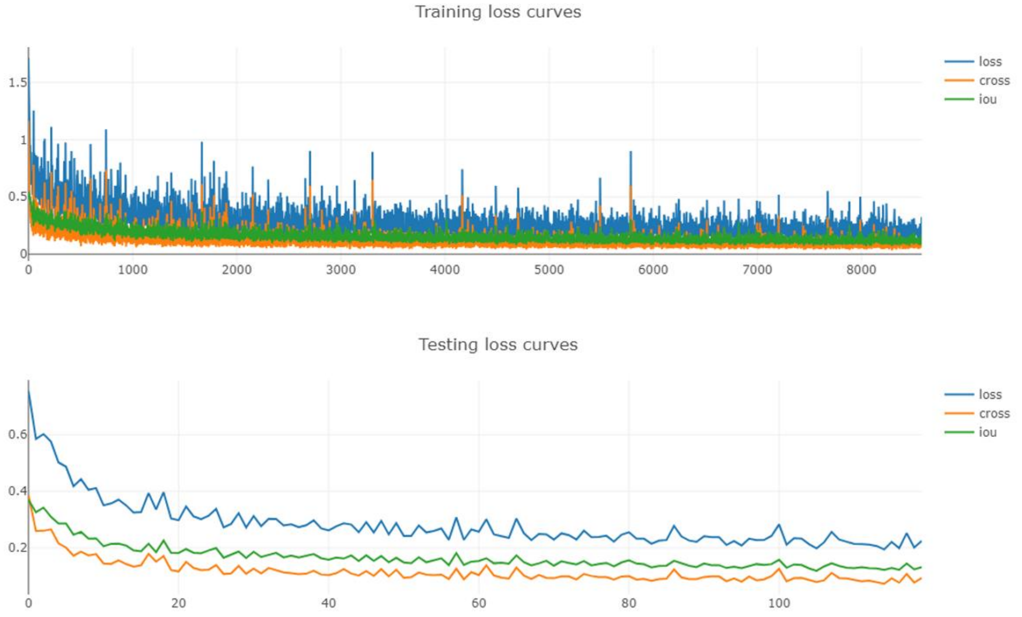

Figure 13. The loss curves of training and testing on rim.

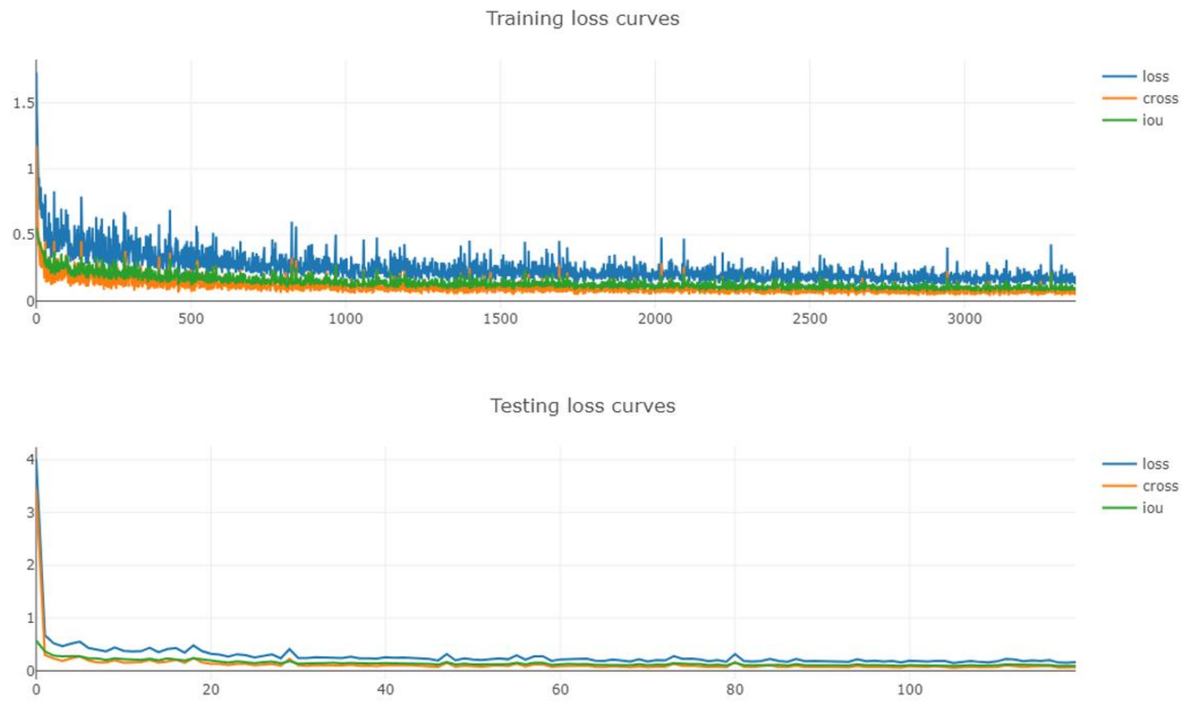

Figure 14. The loss curves of training and testing on drishti.

4.4. Ablation experiments

To evaluate the efficacy of individual modules, we performed ablation experiments on the RIM-ONE-R3 dataset. The results were compared in terms of the multiply-accumulate operations (Macs), parameters (Params), running time (Time), and segmentation accuracy.

Table 5. Results of ablation experiments.

	Macs	Params	Time	DSC_OD	DSC_OC
Remove all MLP modules	1.19 G	0.279 M	11.44 ms	0.9630	0.8620
Replace all modules with SR-MLP	1.35 G	0.514 M	23.32 ms	0.9713	0.9032
Replace all modules with S-MLP	1.35 G	0.514 M	13.93 ms	0.9720	0.8928
Remove the PSP module	1.35 G	0.623 M	18.80 ms	0.9594	0.8881
Remove the first layer of downsampling and the last layer of upsampling	2.15 G	0.486 M	34.60 ms	0.9728	0.8982
Remove the SE-Block module	1.35 G	0.515 M	14.09 ms	0.9663	0.8749
Replace all modules with SE-Block	1.35 G	0.517 M	15.95 ms	0.9736	0.8988
Remove the fuzzy module	1.35 G	0.513 M	**11.05 ms**	0.9713	0.8917
Replace all CNN modules with fuzzy module	1.35 G	0.514 M	84.58 ms	0.9716	0.8972
Proposed	**1.35 G**	**0.513 M**	14.18 ms	**0.9747**	**0.9044**

4.4.1. Effectiveness of MLP modules

To investigate the impact of MLP modules on network performance, we conducted experiments under three scenarios, and the results are presented in Table 5. First, we removed the proposed two MLP modules, which led to faster inference and reduced parameters, as shown in row 1 of Table 5. However, compared to the proposed MLP model, this resulted in a 1.2 and 4.2% decrease in accuracy for the OD and OC regions, respectively. Therefore, despite occupying nearly 50% of the proposed model's parameters, the MLP modules showed significant performance improvements. Next, we replaced the proposed MLP architecture with base SR-MLP and S-MLP models, as respectively shown in rows 2 and 3 of Table 5. Both models had similar inference time and parameters, but in terms of running time and segmentation accuracy, the S-MLP performed slightly better than the SR-MLP. However, both models underperformed compared to the proposed MLP model. Moreover, the SR-MLP module was found to be more effective in handling OD regions with clear fundus contours and fewer lesions and capillaries, while the S-MLP module was better suited for OC regions with less distinct contours and abundant capillaries and large lesion areas. We also observed that the MLP module accounted for almost 50% of the model’s parameters, and that two displacement increased inference time by 67% relative to one displacement operation. The final model utilized a standard MLP module at the lower level to focus on rich details and a displacement MLP module at the higher level to capture a broader range of features, resulting in a more balanced performance. This highlights the efficacy of the proposed MLP module.

4.4.2. Effectiveness of fuzzy module

We evaluated the effectiveness of the proposed fuzzy module as a tool to model uncertainty in retinal fundus images, as well as its impact on segmentation performance. We conducted experiments with two variations of our architecture: one without the fuzzy module and another where each CNN layer was replaced with a fuzzy module. The results are presented in Table 5, rows 8 and 9. In the absence of the fuzzy module, the DSC is lower than for the proposed method and decreased by 0.3 and 1.3% for the OD and OC regions, respectively. When using the fuzzy module instead of all of the CNN modules in the proposed framework, as can be seen in row 9 of Table 5, the segmentation accuracy was slightly improved in the OC region realative to the above method without the fuzzy module; however the OA was still lower than that for the proposed model. These findings demonstrate the efficacy of the fuzzy module in our proposed framework.

4.4.3. Effectiveness of PSP module

As presented in row 4 of Table 5, the PSP module yielded a 1.5 and 1.6% improvements in OD and OC segmentation accuracy, respectively, as compared to the traditional up sampling method. Furthermore, the PSP module reduced the number of parameters by 18% and reduce inference time by 25%. To further demonstrate the efficacy of the PSP module, we removed the first downsampling layer and the last upsampling layer; the results are shown in row 5 of Table 5. Despite a 5% reduction in parameters, increased inference time by 144%, and the accuracy was similar to that for the experiment in row 4. In conclusion, the PSP module reduces the number of parameters; however, it increases the feature map size processed by the module, resulting in computationally intensive inference speed.

Additionally, the PSP module was observed to perform better with high-level features that have smaller sizes, making it more advantageous to use the PSP module for such features.

4.4.4. Effectiveness of SE-Block:

In order to study the contribution of the SE-Block module to the proposed model, we separately tested the performance of each layer with and without the SE-Block module; the results are detailed in rows 6 and 7 of Table 5. The experimental results show that the SE-Block is a lightweight module that has a negligible effect on the inference speed, adding only about 0.1 ms per SE-Block, while requiring a negligible increase in the number of parameters. The results also show that the SE-Block can improve the model's performance, but the gains plateau as the number of layers increases. Specifically, the difference in performance between using the SE-Block once (as in the proposed method, the last row of Table 5) and using it five times (by replacing all modules with SE-Block, as in the seventh row of Table 5) was observed to be marginal.

4.4.5. Computational complexity:

A significant contribution of our model is that the network can be trained efficiently. As noted in [58], network speed is a direct metric for measuring efficiency, whereas the Macs and parameters are indirect metrics. Therefore, not only is the inference time an important evaluation metric, the computational cost and parameters are also important for the evaluation of CNNs. Therefore, to verify the complexity and timeliness of our model, we compared our method with the state-of-the-art methods [17,50,53,54,56] in three aspects: Macs, Params and Time. We re–ran the algorithms with the same division of training and testing; the results are shown in Table 6.

From Table 6, it can be seen that the proposed model not only improves the segmentation accuracy relative to that of the listed methods, it also reduces the computational cost and the number of parameters. Especially, in terms of parameters, the proposed model has significantly fewer parameters than other models, and not even in the same order of magnitude. This means that the proposed model compresses the model and reduces the complexity of the network, making it possible to be deployed on real-time platforms.

Table 6. Comparison of the computational complexity of different methods.

Method	Macs	Params	Time
UNet [50]	54.73 G	31.04 M	20.98 ms
BGA-Net [53]	7.36 G	5.81 M	15.11 ms
GDCSeg [54]	9.36 G	25.71 M	23.49 ms
CE-Net [56]	8.94 G	29 M	19.64 ms
Segtran [17]	148.05 G	172.72 M	65.56 ms
Proposed	**1.35 G**	**0.516 M**	**14.81 ms**

5. Conclusions

In this paper, we have introduced a deep learning-based approach for the simultaneous segmentation of the OD and OC in retinal fundus images. By incorporating fuzzy learning and a lightweight MLP architecture, we have simplified the model, which has resulted in enriched feature extraction and reduced complexity. Our proposed model can also be considered as an extension of traditional deep learning segmentation methods. Experimental results on the DRISGHTI-GS, RIM-ONE-R3, and REFUGE datasets demonstrated promising performance. Compared to other methods, our proposed approach offers faster inference, reduced complexity and fewer parameters, while also achieving state-of-the-art segmentation performance.

Use of AI tools declaration

The authors declare that they have not used artificial intelligence tools in the creation of this article.

Acknowledgments

This work was supported in part by the Shanxi Scholarship Council of China under grant 2023-015.

Conflict of interest

The authors declare that there is no conflict of interest.

References

1. V. S. Mary, E. B. Rajsingh, G. R. Naik, Retinal fundus image analysis for diagnosis of glaucoma: a comprehensive survey, *IEEE Access*, **4** (2016), 4327–4354. https://doi.org/10.1109/access.2016.2596761
2. Y. C. Tham, X. Li, T. Y. Wong, H. A. Quigley, T. Aung, C. Cheng, Global prevalence of glaucoma and projections of glaucoma burden through 2040: a systematic review and meta-analysis, *Ophthalmology*, **121** (2014), 2081–2090. https://doi.org/10.1016/j.ophtha.2014.05.013
3. H. A. Quigley, A. T. Broman, The number of people with glaucoma worldwide in 2010 and 2020, *Br J. ophthalmol.*, **90** (2006), 262–267. https://doi.org/10.1136/bjo.2005.081224
4. M. U. Akram, A. Tariq, S. Khalid, M. Y. Javed, S. Abbas, U. U. Yasin, Glaucoma detection using novel optic disc localization, hybrid feature set and classification techniques, *Australas. Phys. Eng. Sci. Med.*, **38** (2015), 643–655. https://doi.org/10.1007/s13246-015-0377-y
5. M. A. Fernandez-Granero, A. Sarmiento, D. Sanchez-Morillo, S. Jiménez, P. Alemany, I. Fondón, Automatic CDR estimation for early glaucoma diagnosis, *J. Healthc. Eng.*, **2017** (2017), 5953621. https://doi.org/10.1155/2017/5953621
6. S. Wang, L. Yu, X. Yang, C. W. Fu, P. A. Heng, Patch-based output space adversarial learning for joint optic disc and cup segmentation, *IEEE Trans. Med. Imaging*, **38** (2019), 2485–2495. https://doi.org/10.1109/TMI.2019.2899910
7. H. Fu, J. Cheng, Y. Xu, D. W. K. Wong, J. Liu, X. Cao, Joint optic disc and cup segmentation based on multi-label deep network and polar transformation, *IEEE Trans. Med. Imaging*, **37** (2018), 1597–1605. https://doi.org/10.1109/TMI.2018.2791488

8. Z. Zhang, H. Fu, H. Dai, J. Shen, Y. Pang, L. Shao, Et-net: A generic edge-attention guidance network for medical image segmentation, in *Medical Image Computing and Computer Assisted Intervention*, Springer International Publishing, (2019), 442. https://doi.org/10.1007/978-3-030-32239-7_49
9. S. Pathan, P. Kumar, R. Pai, S. V. Bhandary, Automated detection of optic disc contours in fundus images using decision tree classifier, *Biocybern. Biomed. Eng.*, **40** (2020), 52–64. https://doi.org/10.1016/j.bbe.2019.11.003
10. P. S. Mittapalli, G. B. Kande, Segmentation of optic disk and optic cup from digital fundus images for the assessment of glaucoma, *Biomed. Signal Process. Control*, 24 (2016), 34–46. https://doi.org/10.1016/j.bspc.2015.09.003
11. D. W. K. Wong, J. Liu, J. H. Lim, X. Jia, F. Yin, H. Li, et al., Level-set based automatic cup-to-disc ratio determination using retinal fundus images in ARGALI, in *2008 30th annual international conference of the IEEE engineering in medicine and biology society*, (2008), 2266. https://doi.org/10.1109/IEMBS.2008.4649648
12. J. Liu, D. W. K. Wong, J. H. Lim, X. Jia, F. Yin, H. Li, et al., Optic cup and disk extraction from retinal fundus images for determination of cup-to-disc ratio, in 2*008 3rd IEEE Conference on Industrial Electronics and Applications*, (2008), 1828. https://doi.org/10.1109/ICIEA.2008.4582835
13. R. A. Abdel-Ghafar, T. Morris, Progress towards automated detection and characterization of the optic disc in glaucoma and diabetic retinopathy, *Med. Inform. Int. Med.*, **32** (2007), 19–25. https://doi.org/10.1080/14639230601095865
14. F. Mendels, C. Heneghan, J. Thiran, Identification of the optic disk boundary in retinal images using active contours, in *Proceedings of Irish Machine Vision and Image Processing Conference (IMVIP),* (1999), 103.
15. M. Lalonde, M. Beaulieu, L. Gagnon, Fast and robust optic disc detection using pyramidal decomposition and Hausdorff-based template matching, *IEEE Trans. Med. Imaging,* **20** (2001), 1193–1200. https://doi.org/10.1109/42.963823
16. S. Li, X. Sui, X. Luo, X. Xu, Y. Liu, R. Goh, Medical image segmentation using squeeze-and-expansion transformers, preprint, arXiv: 2105.09511.
17. J. Wu, H. Fang, F. Shang, D. Yang, Z. Wang, J. Gao, et al., SeATrans: learning segmentation-assisted diagnosis model via transformer, in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer Nature Switzerland, (2022), 677. https://doi.org/10.1007/978-3-031-16434-7_65
18. T. Yu, X. Li, Y. Cai, M. Sun, P. Li, S2-mlp: Spatial-shift mlp architecture for vision, in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, (2022), 297. https://doi.org/10.1109/WACV51458.2022.00367
19. D. Lian, Z. Yu, X. Sun, S. Gao, As-mlp: An axial shifted mlp architecture for vision, preprint, arXiv: 2107.08391. https://doi.org/10.48550/arXiv.2107.08391
20. I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, et al., Mlp-mixer: An all-mlp architecture for vision, *Adv. Neural Inform. Process. Syst.* **34** (2021), 24261–24272.
21. A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, et al., An image is worth 16x16 words: Transformers for image recognition at scale, preprint, arXiv: 2010.11929. https://doi.org/10.48550/arXiv.2010.11929

22. J. M. J. Valanarasu, V. M. Patel, UNeXt: MLP-based rapid medical image segmentation network, preprint, arXiv: 2203.04967. https://doi.org/10.48550/arXiv.2010.11929
23. H. K. Kwan, Y. Cai, A fuzzy neural network and its application to pattern recognition, *IEEE Trans. Fuzzy Syst.*, **2** (1994), 185–193. https://doi.org/10.1109/91.298447
24. S. Zhou, Q. Chen, X. Wang, Fuzzy deep belief networks for semi-supervised sentiment classification, *Neurocomputing*, **131** (2014), 312–322. https://doi.org/10.1016/j.neucom.2013.10.011
25. R. Zhang, F. Shen, J. Zhao, A model with fuzzy granulation and deep belief networks for exchange rate forecasting, in *2014 international joint conference on neural networks (IJCNN)*, (2014), 366. https://doi.org/10.1109/IJCNN.2014.6889448
26. G. PadmaPriya, K. Duraiswamy, K. Rangasamy, Association of deep learning algorithm with fuzzy logic for multidocument text summarization, *J. Theor. Appl. Inform. Technol.*, **62** (2014).
27. Sun. M, Huang. W, Wang. J. Density-Sorting-Based convolutional fuzzy min-max neural network for image classification, in *2021 International joint conference on neural networks (IJCNN)*, (2021), 1. https://doi.org/10.1109/IJCNN52387.2021.9534394
28. Y. Deng, Z. Ren, Y. Kong, F. Bao, Q. Dai, A hierarchical fused fuzzy deep neural network for data classification, *IEEE Trans. Fuzzy Syst.*, **25** (2016), 1006–1012. https://doi.org/10.1109/TFUZZ.2016.2574915
29. T. Shen, J. Wang, C. Gou, F. Y. Wang, Hierarchical fused model with deep learning and type-2 fuzzy learning for breast cancer diagnosis, *IEEE Trans. Fuzzy Syst.* **28** (2020), 3204–3218. https://doi.org/10.1109/TFUZZ.2020.3013681
30. L. A. Zadeh, Fuzzy Sets, *Inf. Control*, (1965), 338–353. https://doi.org/10.1016/S0019-9958(65)90241-X
31. C. T. Lin, C. S. G. Lee, Neural-network-based fuzzy logic control and decision system, *IEEE Trans. Comput.*, **40** (1991), 1320–1336. https://doi.org/10.1109/12.106218
32. J. J. Buckley, Y. Hayashi, Fuzzy neural networks: A survey, *Fuzzy Sets Syst.*, **66** (1994), 1–13. https://doi.org/10.1016/0165-0114(94)90297-6
33. K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, *Neural Networks*, **2** (1989), 359–366. https://doi.org/10.1016/0893-6080(89)90020-8
34. M. Egmont-Petersen, D. de Ridder, H. Handels, Image processing with neural networks—a review, *Pattern Recogn.*, **35** (2002), 2279–2301. https://doi.org/10.1016/S0031-3203(01)00178-9
35. A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, *Adv. Neural Inform. Processing Syst.*, **25** (2012).
36. C. Farabet, C. Couprie, L. Najman, Y. LeCun, Learning hierarchical features for scene labeling, *IEEE Trans. Pattern Anal. Mach. Intell.*, **35** (2012), 1915–1929. https://doi.org/1915-1929.10.1109/TPAMI.2012.231
37. Z. Yan, H. Zhang, B. Wang, S. Paris, Y. Yu, Automatic photo adjustment using deep neural networks, *ACM Trans. Graph.*, **35** (2016), 1–15. https://doi.org/10.1145/2790296
38. H. Greenspan, B. Van Ginneken, R. M. Summers, Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique, *IEEE Trans. Med. Imaging*, **35** (2016), 1153–1159. https://doi.org/10.1109/TMI.2016.2553401

39. I. Yilmaz, Comparison of landslide susceptibility mapping methodologies for Koyulhisar, Turkey: conditional probability, logistic regression, artificial neural networks, and support vector machine, *Environm. Earth Sci.*, **61** (2010), 821–836. https://doi.org/10.1007/s12665-009-0394-9
40. G. Z. M. Jalali, R. E. Nouri, Prediction of municipal solid waste generation by use of artificial neural network: A case study of Mashhad, *Int. J. Environ. Res*., **2** (2008), 13–22.
41. A. Howard, M. Sandler, G. Chu, L. C. Chen, B. Chen, M. Tan, et al., Searching for mobilenetv3, in *Proceedings of the IEEE/CVF international conference on computer vision*, (2019), 1314. https://doi.org/10.1109/ICCV.2019.00140
42. C. Cui, T. Gao, S. Wei, Y.Du, R. Guo, S. Dong, et al., PP-LCNet: A lightweight CPU convolutional neural network, preprint, arXiv: 2109.15099. https://doi.org/10.48550/arXiv.2109.15099
43. K. He, X. Zhang; S. Ren, J. Sun, Deep residual learning for image recognition, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, (2016), 770. https://doi.org/10.1109/CVPR.2016.90
44. P. Ramachandran, B. Zoph, Q. V. Le, Searching for activation functions, preprint, arXiv: 1710.05941.
45. Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, et al., Swin transformer: Hierarchical vision transformer using shifted windows, in *Proceedings of the IEEE/CVF international conference on computer vision*, (2021), 10012.
46. H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, (2017), 2881. https://doi.org/10.1109/CVPR.2017.660
47. J. Sivaswamy, S. R. Krishnadas, G. D. Joshi, M. Jain, A. U. S. Tabish, Drishti-gs: Retinal image dataset for optic nerve head (onh) segmentation, in *2014 IEEE 11th international symposium on biomedical imaging (ISBI),* (2014), 53. https://doi.org/10.1109/ISBI.2014.6867807
48. F. Fumero, S. Alayon, J. L. Sanchez, J. Sigut, M. Gonzalez-Hernandez, RIM-ONE: An open retinal image database for optic nerve evaluation, in *2011 24th international symposium on computer-based medical systems (CBMS),* (2011), 1. https://doi.org/10.1109/CBMS.2011.5999143
49. J. I. Orlando, H. Fu, J. B. Breda, K. Van Keer, D. R. Bathula, A. Diaz-Pinto, et al., Refuge challenge: A unified framework for evaluating automated methods for glaucoma assessment from fundus photographs, *Med. Image Anal.*, **59** (2020), 101570. https://doi.org/10.1016/j.media.2019.101570
50. O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference*, (2015), 234. https://doi.org/10.1007/978-3-319-24574-4_28
51. S. Wang, L. Yu, X. Yang, C. W. Fu, P. A. Heng, Patch-based output space adversarial learning for joint optic disc and cup segmentation, *IEEE Trans. Med. Imaging*, **38** (2019), 2485–2495. https://doi.org/10.1109/TMI.2019.2899910
52. H. Fu, J. Cheng, Y. Xu, D. W. K. Wong, J. Liu, X. Cao, Joint optic disc and cup segmentation based on multi-label deep network and polar transformation, *IEEE Trans. Med. Imaging*, **37** (2018), 1597–1605. https://doi.org/10.1109/TMI.2018.2791488

53. L. Luo, D. Xue, F. Pan, X. Feng, Joint optic disc and optic cup segmentation based on boundary prior and adversarial learning, *Int. J. Comput. Assist. Radiol. Surgery*, **16** (2021), 905–914. https://doi.org/10.1007/s11548-021-02373-6
54. Q. Zhu, X. Chen, Q. Meng, J. Song, G. Luo, M. Wang, et al., GDCSeg-Net: general optic disc and cup segmentation network for multi-device fundus images, *Biomed. Optics Express*, **12** (2021), 6529–6544. https://doi.org/10.1364/BOE.434841
55. R. Ali, B. Sheng, P. Li, Y. Chen, H. Li, P. Yang, et al., Optic disk and cup segmentation through fuzzy broad learning system for glaucoma screening, *IEEE Trans. Industri. Inform.*, **17** (2020), 2476–2487. https://doi.org/10.1109/TII.2020.3000204
56. Z. Gu, J. Cheng, H. Fu, K. Zhou, H. Hao, Y. Zhao, et al., Ce-net: Context encoder network for 2d medical image segmentation, *IEEE Trans. Med. Imaging*, **38** (2019), 2281–2292. https://doi.org/10.1109/TMI.2019.2903562
57. Z. Zhang, H. Fu, H. Dai, J. Shen, Y. Pang, L. Shao, Et-net: A generic edge-attention guidance network for medical image segmentation, in *22nd International Conference*, Springer International Publishing, (2019), 442–450. https://doi.org/10.1007/978-3-030-32239-7_49
58. N. Ma, X. Zhang, H. T. Zheng, J. Sun, Shufflenet v2: Practical guidelines for efficient CNN architecture design, in *Proceedings of the European conference on computer vision (ECCV)*, (2018), 116. https://doi.org/10.1007/978-3-030-01264-9_8