Fully Convolutional Boundary Regression for Retina OCT Segmentation

Yufan He[1(✉)], Aaron Carass[1,2], Yihao Liu[1], Bruno M. Jedynak[3], Sharon D. Solomon[4], Shiv Saidha[5], Peter A. Calabresi[5], and Jerry L. Prince[1,2]

[1] Department of Electrical and Computer Engineering, The Johns Hopkins University, Baltimore, MD 21218, USA
yhe35@jhu.edu
[2] Department of Computer Science, The Johns Hopkins University, Baltimore, MD 21218, USA
[3] Department of Mathematics and Statistics, Portland State University, Portland, OR 97201, USA
[4] Wilmer Eye Institute, The Johns Hopkins University School of Medicine, Baltimore, MD 21287, USA
[5] Department of Neurology, The Johns Hopkins University School of Medicine, Baltimore, MD 21287, USA

Abstract. A major goal of analyzing retinal optical coherence tomography (OCT) images is retinal layer segmentation. Accurate automated algorithms for segmenting smooth continuous layer surfaces, with correct hierarchy (topology) are desired for monitoring disease progression. State-of-the-art methods use a trained classifier to label each pixel into background, layer, or surface pixels. The final step of extracting the desired smooth surfaces with correct topology are mostly performed by graph methods (e.g. shortest path, graph cut). However, manually building a graph with varying constraints by retinal region and pathology and solving the minimization with specialized algorithms will degrade the flexibility and time efficiency of the whole framework. In this paper, we directly model the distribution of surface positions using a deep network with a fully differentiable soft argmax to obtain smooth, continuous surfaces in a single feed forward operation. A special topology module is used in the deep network both in the training and testing stages to guarantee the surface topology. An extra deep network output branch is also used for predicting lesion and layers in a pixel-wise labeling scheme. The proposed method was evaluated on two publicly available data sets of healthy controls, subjects with multiple sclerosis, and diabetic macular edema; it achieves state-of-the art sub-pixel results.

Keywords: Retina OCT · Deep learning segmentation · Surface segmentation

1 Introduction

Optical coherence tomography (OCT), which uses light waves to rapidly obtain 3D retina images, is widely used in the clinic. Retinal layer thicknesses change

D. Shen et al. (Eds.): MICCAI 2019, LNCS 11764, pp. 120–128, 2019.
https://doi.org/10.1007/978-3-030-32239-7_14

with certain diseases [15] and the analysis of OCT images improves disease monitoring. Manually segmenting the images and measuring the thickness is time consuming, and fast automated retinal layer segmentation tools are routinely used for this purpose.

The goal of layer segmentation is to obtain smooth, continuous retina layer surfaces with the correct anatomical ordering; these results can then be used for thickness analysis or registration [14]. State-of-the-art methods use a trained classifier (e.g, random forest (RF) [13] or deep network [5]) for coarse pixel-wise labeling and then level set [2] or graph methods [3,5,6,13] for surface estimation. Deep networks, which can learn features automatically, have been used in retina layer segmentation. ReLayNet [18] uses a fully convolutional network (FCN) to segment eight layers and edema by classify each pixel into layer or background classes. The main problem with this pixel-wise labeling scheme is that the layer topology is not guaranteed and final continuous and smooth surfaces are not always obtained. Ben-Cohen et al. [1] extract the boundaries from the layer maps using a Sobel filter and then use a shortest path algorithm to extract the surfaces. Other work focuses on classifying the pixels into surface pixels or background. Fang et al. [5] use a convolutional neural network (CNN) and Kugelman et al. [12] use a recurrent neural network (RNN) to classify the center pixel of a patch into boundary pixel or background; then a graph method is used to extract the final surfaces. In order to build a topologically correct graph [6,13], surface distances and smoothness constraints, which are spatially varying, must be experimentally assigned and an energy minimization must be solved outside the deep learning framework. A simpler shortest path graph method [3] was used by [5,12]; however, the shortest path extracts each surface separately, thus the hierarchy is not guaranteed compared to [6,13], especially at the fovea where surface distances can be zero.

The aforementioned graph method cannot be integrated into a deep network, and it needs special parameter settings, which make the method harder to optimize. He et al. [7,9] use a second network to replace the graph method to obtain the smooth, topology-guaranteed surfaces, but this requires much more computation.

In this paper, we propose a new method for obtaining smooth, continuous layer surfaces with the correct topology in an end-to-end deep learning scheme. We use a fully convolutional network to model the position distribution of the surfaces and use a soft-argmax method to infer the final surface positions. The proposed fully convolutional regression method can obtain sub-pixel surface positions in a single feed forward propagation without any fully-connected layer (thus requiring many fewer parameters). Our network has the benefit of being trained end-to-end, with improved accuracy against the state-of-the-art, and it is also light-weight because it does not use a fully-connected layer for regression.

2 Method

Our framework is shown in Fig. 1, we provide a description below for each step. The network has two output branches: the first branch outputs a pixel-wise

labeling segmentation of layers and lesions and the second branch models the distribution of surface positions and outputs the positions of each surface at each column. These two branches share the same feature extractor, a residual U-Net [17]. The input to the network is a three-channel image. One channel is the original flattened OCT image [13], and the other two images are the spatial x and y positions (normalized to the interval $[0, 1]$) of each pixel to provide additional spatial information.

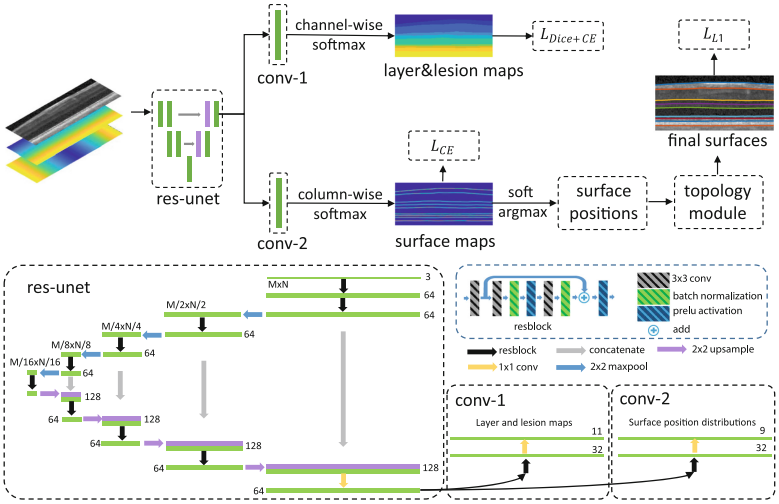

Fig. 1. A schematic of the proposed method (top) and network structure (bottom).

Preprocessing. To reduce memory usage, we flatten a retinal B-scan image to the estimated Bruch's membrane (BM) using an intensity gradient method [13] and crop the retina out.

Surface Position Modeling. Given an image I and a ground truth surface represented by row indexes $x_1^g, \ldots, x_N^g$ across N columns (correspondingly N A-scans), a conventional pixel-wise labeling scheme builds a surface probability map H, where $H(x_j^g, j) = 1$ $(j = 1, \ldots, N)$ and zero else where. A U-Net [17] takes I as input and learns to generate H. Ideally, the surface should be continuous across the image and intersect each column only once. However, the generated probability map may produce zero or multiple positions with high prediction probability in a single column, thus breaking the continuity. Extreme class imbalance between the one-pixel-wide boundary and non-boundary pixels may also cause problems.

In contrast to the pixel-wise labeling scheme, we want to model the surface position distribution $p = p(x_1, \cdots, x_N, I)$ by approximating it with $q = \Pi_{i=1}^{N} q_i(x_i|I; \boldsymbol{\theta})p(I)$. For an input image I, the network generates N independent surface position distributions $q_i(x_i|I; \boldsymbol{\theta}), i = 1, \cdots, N$ at each column. $\boldsymbol{\theta}$ are the

network parameters to be trained by minimizing the K-L divergence between the data distribution p and q. This idea has similarities with Variational Bayesian methods. The direct inference of each x_i from the joint distribution p is hard but it is easier given the simpler independent q_i's. To train the network by minimizing the K-L divergence using the input image I and ground truth x_i^g's, we have

$$\underset{\boldsymbol{\theta}}{\operatorname{argmin}} KL(p||q) = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \iint_{\vec{\boldsymbol{x}},I} p(x_1, \cdots, x_N, I) \log \frac{p(x_1, \cdots, x_N, I)}{\Pi_{i=1}^{N} q_i(x_i|I;\theta)p(I)}, \quad (1)$$

$$= -\underset{\boldsymbol{\theta}}{\operatorname{argmin}} \iint_{\vec{\boldsymbol{x}},I} p(x_1, \cdots, x_N, I) \sum_{i=1}^{N} \log q_i(x_i|I, \boldsymbol{\theta}), \quad (2)$$

$$= -\underset{\boldsymbol{\theta}}{\operatorname{argmin}} \underset{\vec{\boldsymbol{x}},I\sim p}{\mathrm{E}} \left(\sum_{i=1}^{N} \log q_i(x_i|I, \boldsymbol{\theta}) \right). \quad (3)$$

The $\vec{\boldsymbol{x}}$ is a vector of surface positions at N columns and is sampled from data distribution p. In a stochastic gradient descent training scheme, the expectation in Eq. (3) can be removed with a training sample $(I, x_i^g, \cdots, x_N^g)$ and the position x_i^g of the i^{th} column is on the image grid which can only be an integer from 1 to the image row number R. So the loss becomes,

$$\mathcal{L}_{\text{CE}} = -\sum_{i=1}^{N} \sum_{j=1}^{R} \mathbb{1}(x_i^g = j) \log q_i(x_i^g|I, \boldsymbol{\theta}). \quad (4)$$

$\mathbb{1}(x_i^g = j)$ is the indicator function where $\mathbb{1}(x_i^g = j) = 1$ if $x_i^g = j$ and zero elsewhere. Equation (4) is the cross entropy loss for a single surface. In multiple surfaces segmentation, we use the network to output M feature maps (each map has the same size as the input image) for M surfaces. A column-wise softmax is performed independently on these M feature maps to generate surface position probabilities $q_i(x_i|I;\boldsymbol{\theta})$, $i = 1, \ldots, N$ for each column and each surface. An intuitive explanation of the proposed formulation is: instead of classifying each pixel into surfaces or backgrounds, we are selecting a row index at each column for each surface.

Soft-argmax. The deep network outputs marginal conditional distributions $q_i(x_i|i, \boldsymbol{\theta})$ for each column i, so the exact inference of the boundary position at column i from q_i can be performed independently. The differentiable soft-argmax operation (5) used in keypoint localization [10] is used to estimate the final surface position x_i^{sa} at each column i. Thus, we directly obtain the surface positions from the network with soft-argmax,

$$x_i^{sa} = \sum_{r=1}^{R} r q_i(r|I, \boldsymbol{\theta})). \quad (5)$$

We further regularize q_i by directly encouraging the soft-argmax of q_i to be the ground truth, x_i^g, with a smooth $L1$ loss,

$$\mathcal{L}_{\text{L1}} = \sum_{i=1}^{N} 0.5 d_i^2 \, \mathbb{1}(|d_i| < 1) + (|d_i| - 0.5) \, \mathbb{1}(|d_i| \geq 1), \quad d_i = x_i^{sa} - x_i^g \,. \tag{6}$$

Topology Guarantee Module. The layer boundaries within the retina have a strict anatomical ordering. For M surfaces $s_1, \cdots, s_M$ from inner to outer retina, the anatomy requires $s_1(i) \leq s_2(i) \cdots \leq s_M(i), i = 1, \ldots, N$ where $s_j(i)$ is the position of the j^{th} surface at the i^{th} column. The soft-argmax operation produces exact surfaces $s_1, \ldots, s_M$ but may not satisfy the ordering constraint. We update each $s_j(i)$ as $s_j(i)^{new} = s_{j-1}(i) + \text{ReLU}(s_j(i) - s_{j-1}(i))$ iteratively to guarantee the ordering. We implement this as the deep network output layer to guarantee the topology both in the training and testing stages.

Layer and Lesion Pixel-Wise Labeling. Our multi-task network has two output branches. The first branch described above outputs the correctly ordered surfaces whereas the second branch outputs pixel-wise labelings for both layers and lesions. A combined Dice and cross entropy loss [18] is used for the output of this branch. C is the total number of classes, $g_c(x)$ and $p_c(x)$ are the ground truth and predicted probability that pixel x belongs to class c, and Ω_c is the number of pixels in class c and our combined loss is,

$$\mathcal{L}_{\text{Dice+CE}} = \sum_{c=1}^{C} \sum_{x \in \Omega_c} w_c(x) g_c(x) \log(p_c(x)) - \frac{1}{C} \sum_{c=1}^{C} \frac{\epsilon + \sum_{x \in \Omega_i} 2 g_c(x) p_c(x)}{\epsilon + \sum_{x \in \Omega_c} g_c(x)^2 + p_c(x)^2}.$$

Here, $\epsilon = 0.001$ is a smoothing constant and $w_c(x)$ is a weighting function for each pixel [18]. The final network training loss is $\mathcal{L} = \mathcal{L}_{\text{Dice+CE}} + \mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{L1}}$.

3 Experiments

The proposed method was validated on two publicly available data sets. The first data set [8] contains 35 (14 healthy controls (HC) and 21 subjects with multiple sclerosis (MS)) macula Spectralis OCT scans, of which nine surfaces are manually delineated. Each scan has 49 B-scans of size 496×1024. We train on the last 6 HC and last 9 MS subjects and test on the other 20 subjects. The second data set [4] contains 110 B-scans (496×768) from 10 diabetic macular edema patients (each with 11 Bscans). Eight retina surfaces and macular edema have been manually delineated. We train on the last 55 B-scans and test on the challenging first 55 B-scans with large lesions. The network is implemented with Pytorch and trained with an Adam optimizer with an initial learning rate of 10^{-4}, weight decay of 10^{-4}, and a minibatch size of 2 until convergence. The training image is augmented with horizontal flipping and vertical scaling both with probability 0.5.

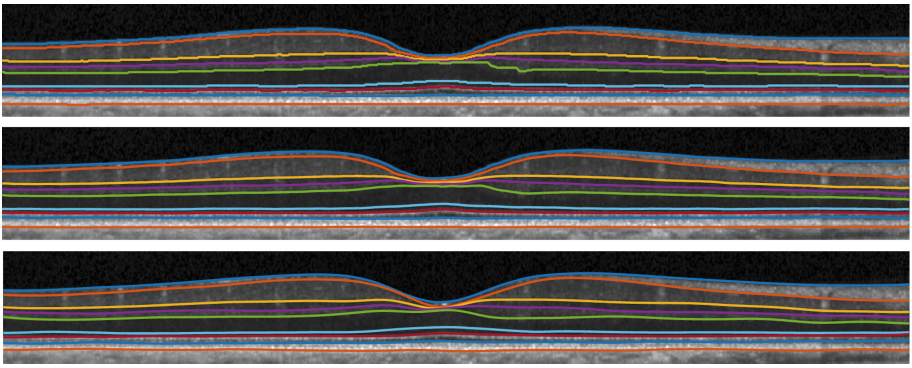

Fig. 2. Visualization of shortest path (top), our result (middle), and manual segmentation (bottom) from a healthy subject.

HC and MS Data Set. We compare our proposed method with several baselines. The AURA tool [13] is a graph based method and it is the state-of-the-art compared to other publicly available tools as shown in [19]. The RNet [9] is a regression deep network that can obtain smooth surfaces from layer segmentation maps. ReLayNet [18] is a variation of U-Net and only outputs layer maps; we obtain the final surface positions by summing up the output layer maps in each column. The shortest path (SP) algorithm [3,5,12] is also compared. We extract the final surfaces using SP on our predicted q_i's (the surface map as shown in Fig. 1). All the baseline methods are retrained on the same data as our proposed method. The mean absolute distances (MADs) for each method are shown in Table 1 (Fig. 2).

DME Data Set. We compare our results with three graph based methods (for final surface extraction): Chiu et al. [4], Rathke et al. [16], and Karri et al. [11]. Since ReLayNet layer surfaces cannot be obtained by simply summing up layer maps (due to lesions), we compare the lesion Dice scores with it. We ignore the positions where Chiu et al. [4]'s result or manual delineation are NaN. The boundary MAD is shown in Table 2. Our network has an extra branch for lesion prediction; however, Rathke's and Karri's method can only output layer surfaces. The Dice score of diabetic macular edema for Chiu's method [4], ReLayNet [18] and ours are 0.56, 0.7, 0.7 respectively. Qualitative results are shown in Fig. 3.

Table 1. Mean absolute distance (MAD) and standard deviation (Std. Dev.) in μm evaluated on 20 manually delineated scans of 9 surfaces, comparing AURA toolkit [13], R-Net [9], ReLayNet [18], SP (shortest path on our surface map), and our proposed method. Depth resolution is 3.9 µm. Numbers in bold are the best in that row.

Boundary	MAD (Std. Dev.)				
	AURA	R-Net	ReLayNet	SP	Our's
ILM	**2.37** (0.36)	2.38 (0.36)	3.17 (0.61)	2.70 (0.39)	2.41 (0.40)
RNFL-GCL	3.09 (0.64)	3.10 (0.55)	3.75 (0.84)	3.38 (0.68)	**2.96** (0.71)
IPL-INL	3.43 (0.53)	2.89 (0.42)	3.42 (0.45)	3.11 (0.34)	**2.87** (0.46)
INL-OPL	3.25 (0.48)	**3.15** (0.56)	3.65 (0.34)	3.58 (0.32)	3.19 (0.53)
OPL-ONL	2.96 (0.55)	2.76 (0.59)	3.28 (0.63)	3.07 (0.53)	**2.72** (0.61)
ELM	2.69 (0.44)	**2.65** (0.66)	3.04 (0.43)	2.86 (0.41)	**2.65** (0.73)
IS-OS	2.07 (0.81)	2.10 (0.75)	2.73 (0.45)	2.45 (0.31)	**2.01** (0.57)
OS-RPE	3.77 (0.94)	3.81 (1.17)	4.22 (1.48)	4.10 (1.42)	**3.55** (1.02)
BM	**2.89** (2.18)	3.71 (2.27)	3.09 (1.35)	3.23 (1.36)	3.10 (2.02)
Overall	2.95 (1.04)	2.95 (1.10)	3.37 (0.92)	3.16 (0.88)	**2.83** (0.99)

Table 2. Mean absolute distance (MAD) in μm for eight surfaces (and the mean of those eight surfaces) evaluated on 55 manually delineated scans comparing Chiu et al. [4], Karri et al. [11], Rathke et al. [16], and our proposed method. Numbers in bold are the best in that column.

	Mean	#1	#2	#3	#4	#5	#6	#7	#8
Chiu	7.82	6.59	8.38	9.04	11.02	11.01	4.84	5.74	5.91
Karri	9.54	**4.47**	11.77	11.12	17.54	16.74	4.99	5.35	**4.30**
Rathke	7.71	4.66	6.78	8.87	11.02	13.60	4.61	7.06	5.11
Ours	**6.70**	4.51	**6.71**	**8.29**	**10.71**	**9.88**	**4.41**	**4.52**	4.61

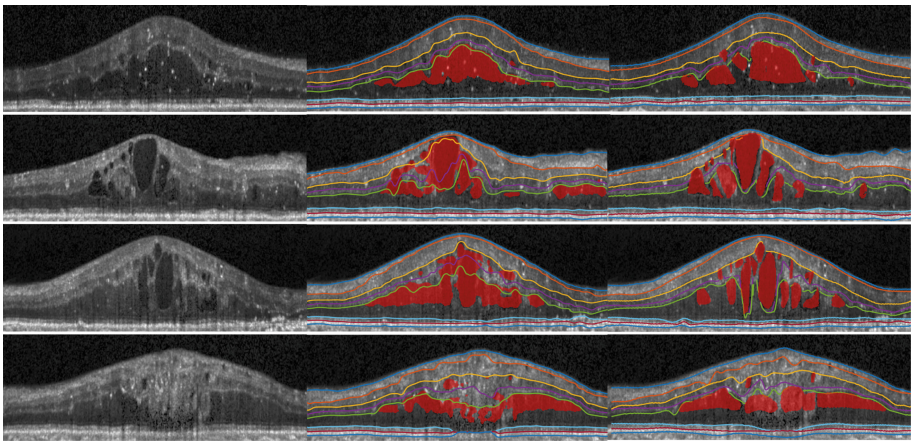

Fig. 3. Visualization of original image (left), our results (middle), manual segmentation (right) of four B-scans from diabetic macular edema subjects.

4 Discussion and Conclusion

In this paper, we proposed a new way for retina OCT layer surface and lesion segmentation without using handcrafted graph based methods. The direct modeling of surface position and the fully differentiable soft-argmax operation generates sub-pixel surface positions in a single feed forward propagation. The generated sub-pixel accuracy surface is smooth and continuous and guarantees correct layer ordering, thus overcoming conventional pixel-wise labeling problems. The topology module affects the network training and thus is not simply a post-processing step. Currently the algorithm works on each B-scan individually, and future work will consider context among different B-scans.

Acknowledgments. This work was supported by the NIH/NEI under grant R01-EY024655.

References

1. Ben-Cohen, A., et al.: Retinal layers segmentation using fully convolutional network in OCT images. In: RSIP Vision (2017)
2. Carass, A., et al.: Multiple-object geometric deformable model for segmentation of macular OCT. Biomed. Opt. Express **5**(4), 1062–1074 (2014)
3. Chiu, S.J., et al.: Automatic segmentation of seven retinal layers in SDOCT images congruent with expert manual segmentation. Opt. Express **18**(18), 19413–19428 (2010)
4. Chiu, S.J., et al.: Kernel regression based segmentation of optical coherence tomography images with diabetic macular edema. Biomed. Opt. Express **6**(4), 1172–1194 (2015)
5. Fang, L., et al.: Automatic segmentation of nine retinal layer boundaries in OCT images of non-exudative AMD patients using deep learning and graph search. Biomed. Opt. Express **8**(5), 2732–2744 (2017)
6. Garvin, M.K., et al.: Automated 3-D intraretinal layer segmentation of macular spectral-domain optical coherence tomography images. IEEE Trans. Med. Imag. **28**(9), 1436–1447 (2009)
7. He, Y., et al.: Towards topological correct segmentation of macular OCT from cascaded FCNs. In: Cardoso, M.J., et al. (eds.) FIFI/OMIA -2017. LNCS, vol. 10554, pp. 202–209. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67561-9_23
8. He, Y., et al.: Retinal layer parcellation of optical coherence tomography images: data resource for multiple sclerosis and healthy controls. Data Brief **22**, 601–604 (2018)
9. He, Y., et al.: Topology guaranteed segmentation of the human retina from OCT using convolutional neural networks. arXiv preprint arXiv:1803.05120 (2018)
10. Honari, S., et al.: Improving landmark localization with semi-supervised learning. In: The IEEE Conference on Computer Vision and Pattern Recognition (2018)
11. Karri, S., et al.: Learning layer-specific edges for segmenting retinal layers with large deformations. Biomed. Opt. Express **7**(7), 2888–2901 (2016)
12. Kugelman, J., et al.: Automatic segmentation of oct retinal boundaries using recurrent neural networks and graph search. Biomed. Opt. Express **9**(11), 5759–5777 (2018)

13. Lang, A., et al.: Retinal layer segmentation of macular OCT images using boundary classification. Biomed. Opt. Express **4**(7), 1133–1152 (2013)
14. Lee, S., et al.: Atlas-based shape analysis and classification of retinal optical coherence tomography images using the functional shape (fshape) framework. Med. Image Anal. **35**, 570–581 (2017)
15. Medeiros, F.A., et al.: Detection of glaucoma progression with stratus OCT retinal nerve fiber layer, optic nerve head, and macular thickness measurements. Invest. Ophthalmol. Vis. Sci. **50**(12), 5741–5748 (2009)
16. Rathke, F., Desana, M., Schnörr, C.: Locally adaptive probabilistic models for global segmentation of pathological OCT scans. In: Descoteaux, M., Maier-Hein, L., Franz, A., Jannin, P., Collins, D.L., Duchesne, S. (eds.) MICCAI 2017. LNCS, vol. 10433, pp. 177–184. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66182-7_21
17. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
18. Roy, A.G., et al.: Relaynet: retinal layer and fluid segmentation of macular optical coherence tomography using fully convolutional networks. Biomed. Opt. Express **8**(8), 3627–3642 (2017)
19. Tian, J., et al.: Performance evaluation of automated segmentation software on optical coherence tomography volume data. J. Biophotonics **9**(5), 478–489 (2016)