Semantic segmentation of artery-venous retinal vessel using simple convolutional neural network

Semantic segmentation of artery-venous retinal vessel using simple convolutional neural network

W Setiawan[1*], M I Utoyo[2], R Rulaningtyas[3], A Wicaksono[4]
[1,2]Mathemathics Department, University of Airlangga, Kampus C Mulyorejo, Surabaya, Indonesia
[3] Physics Department, University of Airlangga, Kampus C Mulyorejo, Surabaya, Indonesia 1Informatics Department, University of Trunojoyo Madura, PO BOX 2 Kamal, Bangkalan, Indonesia
[4]Faculty of nursing & medical, Ramathibodi Hospital, Mahidol University, Thailand

E-mail: riries-r@fst.unair.ac.id

Abstract. Semantic segmentation is how to categorize objects in an image based on pixel color intensity. There is an implementation in the medical imaging. This article discusses semantic segmentation in retinal blood vessels. Retinal blood vessels consist of artery and vein. Arteryvenous segmentation is needed to detect diabetic retinopathy, hypertension, and artherosclerosis. The data for the experiment is Retinal Image vessel Tree Extraction (RITE). Data consists of 20 patches with a dimension of 128 × 128 × 3. The process for performing semantic segmentation consists of 3 method, create a Conventional Neural Network (CNN) model, pre-trained network, and training the network. The CNN model consists of 10 layers, 1 input layer image, 3 convolution layers, 2 Rectified Linear Units (ReLU), 1 Max pooling, 1 transposed convolution layer, 1 softmax and 1 pixel classification layer. The pre-trained network uses the optimization algorithm Stochastic Gradient Descent with Momentum (SGDM), Root Mean Square Propagation (RMSProp) and Adaptive Moment optimization (Adam). Various scenarios were tested to get optimal accuracy. The learning rate is 1e-3 and 1e-2. Minibatch size are 4,8,16,32,64, and 128. The maximum value of epoch is set to 100. The results show the highest accuracy of up to 98.35%

1. Introduction

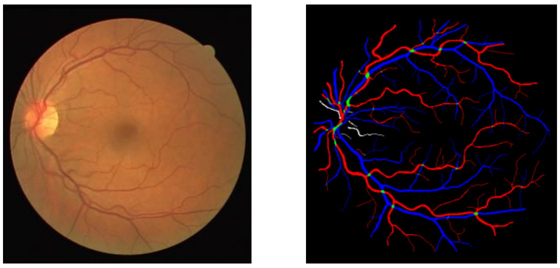
Segmentation is how to categorize objects based on the intensity of color in an image. Semantic segmentation is a segmentation in deep learning using pixel color intensity. Semantic segmentation method can be supervised and unsupervised. If an image has ground truth, the semantic segmentation method is supervised. Vice versa, if there is no ground truth, it is unsupervised.

The application of semantic segmentation includes autonomous car drivers. The system is built to provide knowledge and understanding of the environment. The system can recognize other objects such as motor vehicles, bicycles, roads, traffic signs, trees and other objects commonly found on highways [1]–[3]. Semantic segmentation also develops to diagnose medical images. This helps radiologists to analyze a medical image [4]–[6].

The semantic segmentation system in deep learning needs to build a network. Commonly, network used is Convolutional Neural Network (CNN). There is a various model of CNN, including AlexNet [7], GoogleNet [8] dan ResNet [9]. In addition, there are still many variations on the development of the ImageNet Award winning model. ImageNet is a database of millions of images with thousands of classes. Every year holds a competition to create models with the best accuracy [10].

The development models of CNN include VGG16, VGG19 [11], Resnet50, Resnet 110, Inception V3 and Inception V4 [12]. The difference between these models is the number and the arrangement of the layers. Generally, the model consists of a convolution layer with stride and padding, Rectified Linear Unit (ReLU), Pooling layer, Fully Connected Layer with softmax as an activation function [13]. In addition, to building networks for semantic segmentation, pretraining networks are also carried out. A gradient descent optimization algorithm is applied. Next, the performance measure using accuracy.

This article discusses the application of semantic segmentation to the artery-venous retinal vessel fundus. The system can distinguish an artery and vein in retinal images. The final part of the system can calculate the accuracy and time consumed. The ground truth consists of 5 segments include background, artery, vein, both and uncertain. Blue is an artery, red is a vein, white means uncertain, green indicates that the pixel is passed by both artery-venous. Black is a background. Semantic segmentation is limited to two categories, artery, and vein. The fundus image and the ground truth are shown in Figure 1.



(a) (b)

Figure 1. (a) original image (b) ground truth from RITE dataset [14]

Artery-venous are blood vessels in the retina. If another blood vessel is found in the retina, it is called neovascularization. The characteristics of artery and vein retinal vessels are shown in Table 1 [15]

Table 1. The difference between artery and vein

Artery	Vein
Brighter	Darker
Thinner	Thicker
Central reflex is wider	Small central reflex
2 artery on the optic disc	1 vein on the optic disc

There have been previous studies reported the results of arteryvein segmentation using machine learning. Malek & Tourki carried out the classification of pixel artery-venous blood vessels. The methods used are matched filter, Neural Network, and Principal Component Analysis. Data used for testing are DRIVE and STARE datasets. The test results showed accuracy up to 95.32% [16]. Joshi et al. segment artery-veins using vessel trees and graph search. Test data using DERIVA dataset. Classification uses Fuzzy C-Means [17]. Maheswary & Anandhi conducted a survey of 3 methodologies used to segment artery-veins. First, using image processing techniques, image enhancement, segmentation, thinning, feature extraction, classification, and postprocessing. The second uses the 4 region segmentation in the retinal image, the superior temporal, inferior temporal, superior nasal and nasal inferiors. They do segmentation with graph analysis [15].

Hereafter, Hatami & Goldbaum, detected artery-venous using Local Binary Pattern (LBP). Pre-processing uses Matched Response Filters. Data using STARE dataset. Result shows recognition rate 79.3%. Alam et al. segmenting artery-venous using Optical Density Ratio (ODR) and blood vessel tracking. Data for an experiment is primary data from local hospitals. The results show an accuracy of up to 97.06% [18]. Konderman et al. segmenting artery-venous blood vessels. The vessel profile based feature extraction and the region of interest based vessel. Classification method uses Multi-Layer Perceptron Neural Network and Support Vector Machine. Accuracy results show 95.32% suitability based on pixels [19].

The method of that studies uses machine learning and conventional image processing. Today, deep learning has developed, which is capable of processing big data with relatively fast processing times. The machine used for processing based on Graphical Processing Unit (GPU). Girad and Cheriet segmentation of arteryvenous using the Convolutional neural network (CNN) and likelihood Score Propagation. The data using DRIVE and MESSIDOR with patches size 128x128. The results shows an accuracy up to 93.3% [20]. Furthermore, Ventura et al. doing artery-vein segmentation using Iterative Deep Learning with dealineations and graph algorithms. The CNN Network Model uses VGG. The precision obtained up to 86.1% [21].

Welikala et al. use CNN to detect artery-venous. Parameters used include patches sizes of 25x25, minibatch 128, momentum 0.9 and L2 Normalization 0.0005. Network model consists of 20 layers. The layer consists of 3 convolution layer, 3 Cross chan normalization, 2 max pooling, 5 ReLU, 3 Fully Connected Layer, 1 Dropout, 1 softmax, 1 input layer, and 1 classification layer. Test data was obtained from local data in the UK Biobank. The results show an accuracy up to 91.99% [22]. Hemelings et al segmented arteryvenous using fully convolution network with the U-Net model. Pre-processing uses contrast enhancement. The dataset uses DRIVE, the accuracy shows between 94.11% to 94.42% [23]. From the previous studies is possible to improve accuracy results by modifying existing network models. The novelty of this article is modfying the network model and increasing the performance measure through a higher level of accuracy than previous studies.

2. Research method

The research consists of testing data processing, data patches storing and data labeling, creating semantic segmentation networks, pre-trained networks, train networks and performance measures with accuracy. The research methodology is shown in Figure 2.

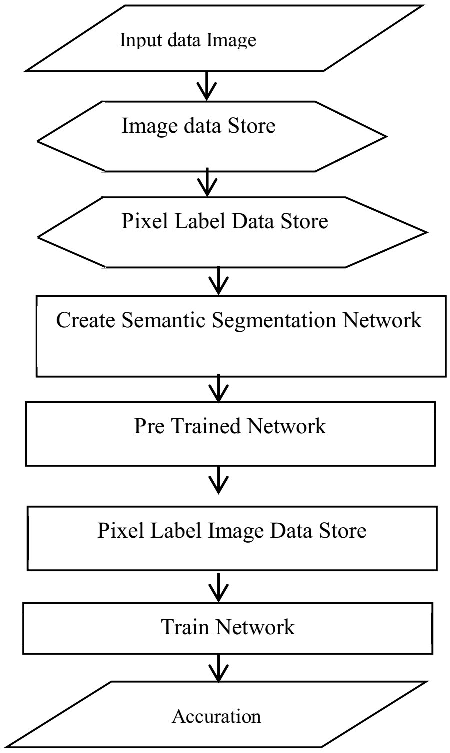
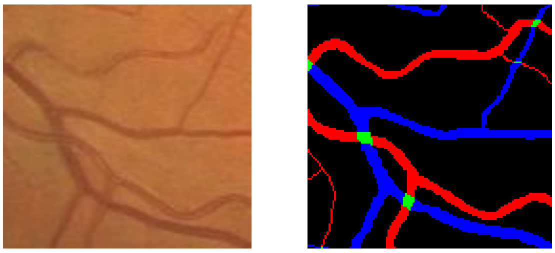


Figure 2. The Research methodology flowchart

2.1. Images Data input

The experiments are carried out in various scenarios. The data using images from the Retinal Image vessel Tree Extraction (RITE). RITE is actually a DRIVE dataset that has been widely used for retinal blood vessel segmentation. RITE was developed by the University of Iowa and Carver College of Medicine. The data experiment using patches or the part of image. Patches are taken by manually cropping any surface of the retinal fundus. The ground truth is crop-ping manually with sizes x_1, $x_2, y_1,$ and y_2 equal to the size of the patches. The representation of patches from the RITE image is shown in Figure 3 [14]



(a) (b)

Figure 3. Patches took from RITE dataset

2.2. Images Datastore and Pixel Label Datastore

Image datastore and datastore pixel label are data patches and labeling storage. Data patches and labeling storage are carried out separately. The number of images in each datastore is 20 patches and 20 ground truth.

2.3. Create Semantic Segmentation Network

Semantic segmentation network is built with Convolutional Neural Network (CNN) model. CNN is the type of network that is most often used for classification purposes or segmentation in deep learning. The advantages of using CNN include:

1. CNN is a network that produces feature learning. The conventional neural network cannot do this.
2. CNN performs weight sharing so that the number of parameters used can be decreased. For example, if CNN has 10 filters with a size of 3 × 3. The number of parameters weighing 3 × 3 × 10 and 10 can be 3 × 3 × 10 + 10 = 100.
3. CNN can perform various functions such as classification, segmentation, image recognition, and image enhancement.
4. CNN can do transfer learning by modifying and using existing models, without learning from the beginning part.

The CNN architecture used consists of convolutional layers, padding, stride, max-pooling, rectified linear unit (ReLU), transposed convolution or deconvolution, softmax, and pixel classification layer. The convolutional layer is a dot product operation between input image matrix and filter matrix. The result of Convolutional Layer will reduce the dimensions of the input layer. Input layer has dimension 128×128×3. The dimension of convolutional layer 64×3×3. Furthermore, there are strides and padding. A stride is a number of shift values carried out during the convolution process. Value of shift can be determined by the user. While the padding is the output /edge of the input image that is not in accordance with the matrix filter. Furthermore, the value on the edge matrix is changed to zero (zero padding) or fixed all values (valid padding). The next layer is Rectified Linear Unit (ReLU). The function of ReLU is change output. If the output is negative, then ReLU changes it to zero. Next, the pooling layer performs dimension reduction on the image matrix. There are three types of pooling layers, max-pooling, average-pooling, and sum-pooling. Maxpooling is the most commonly used. Max-pooling used is 2 × 2 with stride 2 and pooling 0.

The next layer is Transposed Convolution Layer with a dimension of 64 × 4 × 4 with stride 2 and cropping output 1. The softmax have functioned as an activation layer. The final layer is the pixel Classification. Proposed CNN Architecture is found in table 1

Table 2. Proposal CNN Architecture

Layer	Size	Stride	Padding	Output cropping
Image input	$128 \times 128 \times 3$	-	-	-
Convolution	$64 \times 3 \times 3$	x	1	-
ReLU	-	-		-
maxPooling	2×2	2	0	-
Convolution	$64 \times 3 \times 3$	1	1	-
ReLU	-	-	-	-
deConv	$64 \times 4 \times 4$	2	0	1
Convolution	$2\times 1 \times 1$	1	0	-
Sofmax	-	-	-	-
pixelClassification	2	-	-	-

The methodology of semantic segmentation network use CNN include:

1. Initialize the input layer image with the specified size and format.
   Input size [128 128 3]
2. Create Downsampling Network
   Downsampling layer include a convolutional layer, Rectified Linear Unit (ReLU) and max-Pooling.
   ```
   downsamplinglayer {
               Convolutional layer
               ReLU
               maxPooling
               Convolutional layer
               ReLU
               Max-Pooling }
   ```
3. Create Upsampling Network
   Upsampling network consists of transposed convolution layer and ReLU. Transposed Convolution layer also is known as deconvolution layer.
   ```
   Upsamplinglayer {
               Transposedconvolution layer
               ReLU
               Transposedconvolution layer
               ReLU }
   ```
4. Create a pixel classification layer.
   Classification layer consists of softmax and pixel classification layer
5. Sort all layers to complete the semantic segment network
   ```
   Net  = [
         imgLayer
         downsamplingLayer
         UpsamplingLayer
         classificationLayer
         ]
   ```

2.4. Pre-Trained Network

At the pre-trained network, initialization parameters are used for the training phase. Parameters used include the stochastic gradient descent with momentum (SGDM) optimization algorithm, Root Mean

Square Propagation (RMSProp) and Adaptive Optimization (ADAM). The next parameter is the value of learning rate, max Epoch, minibatch size. For other parameters such as the first moment, second moment, and epsilon the default is set to 0.9, 0.999, and 1e-08

2.4.1. Stochastic Gradient Descent with momentum (SGDM)

The Stochastic Gradient Descent is an optimization algorithm to reach the optimal point using the learning rate of the one data at the time of processing. The gradient descent is to use the entire data at one time of processing. The momentum variable serves to accelerate the pace of the learning rate.
The learning rate must be properly initialized. Generally set to 0.001. If the learning rate is too small then the learning time is longer. If it is too large, it can cause overfitting [24]. The cycle of learning movement towards the optimal point is shown in Figure 4.

2.4.2. Root Mean Square Propagation (RMSProp)

RMSProp is a development optimization algorithm from SGDM. If on SGDM, the learning rate is static. RMSProp is adaptive, changes according to conditions. The variables in the RMSProp are shown in equations 1 to 5 [25]

$$E[g^2]_t = \rho E[g^2]_{t-1} + (1-\rho)_{a_{t^2}} \quad (1)$$

$$RMS(g_t) = \sqrt{E[g^2]_t + \varepsilon} \quad (2)$$

$$RMS\ [x]_{t-1} = \sqrt{E[w^2]_{t-1} + \varepsilon} \quad (3)$$

$$w_t = RMS[x]_{t-1}.RMS(g_t)^{-1} g_t \quad (4)$$

$$E[w^2]_t = \rho E[w^2]_{t-1} + (1-\rho)_{w_{t^2}} \quad (5)$$

Herewith :
$E[w^2]_t$ = update gradient accumulator at time t
$E[g^2]_t$ = gradient accumulator at time t
g_t = gradient at time t
ρ = decay rate coeffisien (0.95)
ε = constanta (1e-6)
w_t = weight update at time t
$RMS(g_t)$ = the mean square root of the gradient accumulator at time t
$RMS[x]_{t-1}$ = the mean square root of the update accumulator gradient at time $t-1$

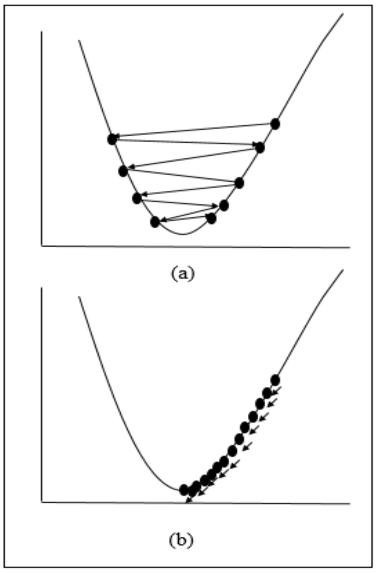


Figure 4: The movement of learning rate (a) large learning rate caused overfitting (b) small learning rate

2.4.3. Adaptive Moment Optimization (ADAM)

Adam is a combination of the AdaGrad optimization algorithm with RMSProp. For several experiments, Adam produced more optimal accuracy than other gradient descent algorithms. Variables in Adam are shown in terms of 6-10 [26].

Herewith,

$$m_t = \beta_1 \times m_{t-1} + (1 - \beta_1) \times g_t \tag{6}$$

$$v_t = \beta_2 \times v_{t-1} + (1 - \beta_2) \times g_t^2 \tag{7}$$

$$\widehat{m_t} = m_t(1 - \beta_1^t)^{-1} \tag{8}$$

$$\widehat{v_t} = v_t(1 - \beta_2^t)^{-1} \tag{9}$$

$$\theta_t = \theta_{t-1} \times \alpha \times \widehat{m_t}\left(\sqrt{\widehat{v_t}} + \varepsilon\right)^{-1} \tag{10}$$

Herewith,

m_t = the first-moment estimation at time t
v_t = the second-moment estimation at time t
$\widehat{m_t}$ = the first-moment estimate is corrected bias at time t
$\widehat{v_t}$ = the second-moment estimate is corrected bias at time t
θ_t = update weight at time t
β_1, β_2 = exponential decay rate

2.5. Pixel Label Image Datastore

The Pixel label Image datastore is matching each pixel in the image datastore and labeling the datastore. Value information for each pixel will be stored in a separate place.

2.6. Train Network

The Train Network is carried out by simultaneously running the Image Datastore pixel label, the semantic segmentation network and pre-trained Network with gradient descent optimization algorithms. In general, the training algorithm is shown in Figure 5.

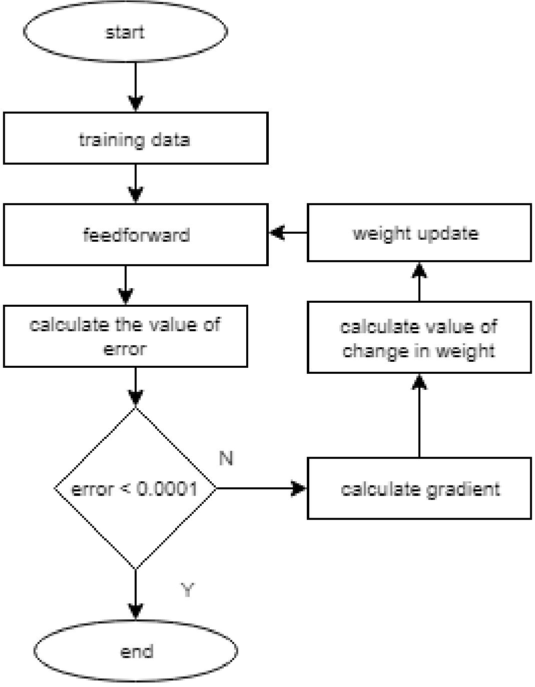


Figure 5. Training algorithm flowchart

3. Result and Discussion

The experiments are carried out on a single Graphical Processing Unit (GPU) with specifications of the Core i7 7700, RAM, Z270 motherboard, GTX 1060 6 Gigabyte D5 amp, DDR 4 16 Gb, PSU 550w, and 60 Gigabyte SSD.

The scenario uses a patch data size of 128x128. Learning rate values are 1e-3 and 1e-2. The size minibatch used is 4,8,16,32,64,128. Optimization algorithm using SGDM, RMSProp and ADAM. The results obtained optimal accuracy of 98.35% using the Adam algorithm with a learning rate of 1e-2 and minibatch size 4. The experiment results are shown in Table 2.

Table 3. Performance of accuration using max Epoch 100

Parameter	Optimization Algorithm		
	SGDM	RMSProp	Adam
α = 1e-3, MB = 64	95.36	27.6	55.14
α = 1e-2, MB = 64	97.87	10.66	97.87
α = 1e-2, MB = 32	97.87	69.42	79.30
α = 1e-2, MB = 16	89.5	97.41	20.27
α = 1e-2, MB = 8	97.78	89.46	77.82
α = 1e-2, MB = 4	96.48	97.70	**98.35**
α = 1e-2, MB = 128	97.87	97.80	88.58
Average	96.10	70.01	73.90

The epoch is between 1 to 100. There is a more optimal accuracy shown in table 3. The parameters set are the same as the experiment in table 2. Accuracy results can be reached up to 100% in the RMSProp algorithm with learning rate 1e-2 and minibatch size 4. The highest results of all accuracy in all epochs are in table 3.

Table 4. The Highest performance of accuration using Epoch 1 - 100

Parameter	Optimization Algorithm		
	SGDM	RMSProp	Adam
α = 1e-3, MB = 64	95.36	60.62	64.36
α = 1e-2, MB = 64	97.87	97.87	97.87
α = 1e-2, MB = 32	97.87	97.41	79.30
α = 1e-2, MB = 16	89.5	80.76	78.15
α = 1e-2, MB = 8	97.78	89.46	92.32
α = 1e-2, MB = 4	96.48	**100**	98.35
α = 1e-2, MB = 128	97.87	97.87	88.58
Average	96.10	89.14	85.56

α = learning rate, MB = Minibatch Size

The use of 10 layer CNN consisting of input layer, downsampling, upsampling, and pixel classification layer can perform semantic segmentation with high accuracy results. Furthermore, the SGDM, RMSProp and Adam optimization algorithms and variations in the learning rate, minibatch size, and epoch values also affect the percentage of accuracy. From the results show the highest average accuration obtained on the SGDM algorithm is 96.10%. Besides accuracy, there are also measurements of the time to run semantic segmentation. The time needed for program execution is 9-14 seconds. When compared to a single Central Processing Unit (CPU) takes 11-17 minutes.

Furthermore, the comparison between the results of accuracy achieved with previous research is shown in table 4. Semantic segmentation built using simple layers can produce accuracy better than previous studies.

Table 5. Comparison with previous research

Author	Dataset	Data type & size	Network	Acc.
Girard & Cheriet [20]	DRIVE, MESSIDOR	Patches 128x128	CNN, likelihood score propagation	93.3%
Ventura et al.[21]	DRIVE	Patches 64x64	VGG	86.1%
Welikala et al.[22]	UK Biobank	Patches 25x25	CNN 20 layer	91.94%
Hemeling et al.[23]	DRIVE	Uncertained	Unet, Fuzzy C Means	94.42%
Proposed method	RITE	Patches 128x128	CNN 10 layer	98.35%

4. Conclusion

In this article semantic segmentation has been carried out on artery-venous vessel image. The experiment using patches from the RITE dataset. Create a system of semantic segmentation, it is necessary to develop a network that is a CNN model, optimization algorithm and training the network. Accuracy is obtained by comparing patches with ground truth. For further research, it can use the full image on the RITE dataset, create a new CNN model and develop an optimization algorithm .

Acknowledgement
We would like to acknowledge the Ministry of Research, Tecnology and Higher Education, Indonesia for funding this research. We also thanks to RITE dataset as experiment data.

References

[1] Treml M et al 2016 Speeding up Semantic Segmentation for Autonomous Driving,” 29th *Conf. Neural Inf. Process. Syst. (NIPS)* 1–7.
[2] Chen B K., Gong C and Yang J 2017 Importance-aware semantic segmentation for autonomous driving system *IJCAI Int. Jt. Conf. Artif. Intell* 1504–1510.
[3] Siam M, Elkerdawy S. and Jagersand M 2018 *Deep Semantic Segmentation for Automated Driving Taxonomy* (Roadmap and Challenges).
[4] G. Wang et al 2017 Interactive Medical Image Segmentation using Deep Learning with Image-specific Fine-tuning 1–11.
[5] Roth H R et al 2018 Deep learning and its application to medical image segmentation 1–6.
[6] Van D S P and Group V 2017 CNN-based Segmentation of Medical Imaging Data
[7] A. Krizhevsky and G. E. Hinton ImageNet Classification with Deep Convolutional Neural Networks 1–9.
[8] Szegedy C et al 2014 Going Deeper with Convolutions 1–9.
[9] He K and Sun J Deep Residual Learning for Image Recognition 1–9.
[10] Li F F and Russakovsky o 2013 Analysis of Large Scale Visual.
[11] Simonyan K and Zisserman A 2014 Very Deep Convolutional Networks for Large-Scale Image Recognition 1–14.
[12] Szegedy C, Ioffe S, Vanhoucke V and Alemi A A 2017 the Impact of Residual Connections on Learning *Proc. Thirty-First AAAI Conf. Artif. Intell. Inception* 4278–4284.
[13] A. Ng et al 2015 *Deep Learning Tutorial* (Univ. Stanford).
[14] Hu Q, Abràmoff M D, and Garvin M K 2015 Automated construction of arterial and venous trees in retinal images *J. Med. Imaging* **2** (4) 044001.
[15] Maheswari S and Anandhi S V 2016 C Lassification Of R Etinal V Essels Into A Rteries And V

EinsA Survey **4**(6) 69–78.

[16] Malek J and Tourki R 2013 Blood vessel classification into arteries and veins in retinal images *Int. Multi-Conference Syst. Signals Devices* 1–6.

[17] Joshi V S, Reinhardt J M, Garvin M K. And Abramoff M D 2014 Automated method for identification and artery-venous classification of vessel trees in retinal vessel networks *PLoS One* **9** (2) 1–12.

[18] Alam M, Son T, Toslak D, Lim J I and Yao X 2018 Combining ODR and Blood Vessel Tracking for Artery–Vein Classification and Analysis in Color Fundus Images *Transl. Vis. Sci. Technol* **7** (2) 23.

[19] Kondermann C, Kondermann D and Yan M 2017 Blood vessel classification into arteries and veins in retinal images 651247.

[20] Girard F and Cheriet F 2017 Artery / Vein Classification In Fundus Images Using Cnn And Likelihood Score Propagation,Canada 720–724.

[21] Ventura C, Pont-Tuset J, Caelles S, Maninis K.K. and Van Gool L 2017 Iterative Deep Learning for Network Topology Extraction.

[22] Welikala R A et al 2017 Automated arteriole and venule classification using deep learning for retinal images from the UK Biobank cohort *Comput. Biol. Med* **90** 23–32.

[23] Hemelings R, Stalmans I, Van Keer K, Elen B and De Boever P 2018 Extended Abstract : Artery-Vein Segmentation in Fundus Images using a Fully Convolutional Network (i) 3–5.

[24] Ruder S 2016 An overview of gradient descent optimization 1–14.

[25] Hinton G E, Srivastava N and Swersky K 2012 Lecture 6a- overview of mini-batch gradient descent *COURSERA Neural Networks Mach. Learn* 31.

[26] Kingma D P and Ba J. 2014 Adam: A Method for Stochastic Optimization1–15.