Interpretability on Deep Retinal Image Understanding Network

Manal AlGhamdi

Department of Computer Science and Artificial Intelligence, University of Umm AL-Qura, Makkah, Saudi Arabia

Abstract
In the last 10 years, artificial intelligence (AI) has shown more predictive accuracy than humans in many fields. Its promising future founded on its great performance increases people's concern about its black-box mechanism. In many fields, such as medicine, mistakes lacking explanations are hardly accepted. As a result, research on interpretable AI is of great significance. Although much work about interpretable AI methods are common in classification tasks, little has focused on segmentation tasks. In this paper, we explored the interpretability on a Deep Retinal Image Understanding (DRIU) network, which is used to segment the vessels from retinal images. We combine the Grad Class Activation Mapping (Grad-CAM), commonly used in image classification, to generate saliency map, with the segmentation task network. Through the saliency map, we got information about the contribution of each layer in the network during predicting the vessels. Therefore, we adjusted the weights of last convolutional layer manually to prove the accuracy of the saliency map generated by Grad-CAM. According to the result, we found the layer 'upsample2' to be the most important during segmentation, and we improved the mIoU score (an evaluation method) to some extent.

Keywords:
Retinal image synthesis, generative adversarial networks, image-to-image translation, medical image segmentation.

1. Introduction

Deep learning achieved astonishing success in computer vision during the last few years because of its great performance. However, there is still a long way to go before widespread use in some fields, such as medicine. In those fields, failure is intolerant given the connection between disease diagnosis and living quality, or worse, fatality. Therefore, it is hard to adopt algorithms with poor "explainability" or "interpretability" just because its high black-box performance [21]. According to [16, 5, 3, 9], although machine learning (ML) and neural networks (NN) show promising performance in several medical tasks, they are far from perfect.

As a result, interpretability has become an urgent problem that appeals to many researchers. It includes a series of issues: Do we know where it goes against our expectation if the prediction is wrong? Can we understand further how the algorithm works if its performance is beyond specialists? What are the most important parts in the network leading to its prediction? To figure out those questions, researchers focus on such aspects that (1) to explain the decisions made by algorithms, (2) to expose the patterns of the inner mechanism of a network, (3) to add some coherent models and demonstrate them with more mathematics. However, there typically is a trade-off between performance and interpretability: complex, high-performance networks like deep residual networks (ResNets) [7] have a huge number ($L > 200$) of layers. It is challenging to explore how these layers work and connections between them, and we therefore choose to start with a relatively small network for Deep Retinal Image Understanding (DRIU) presented in [13]. We explored how the network segment the vessels from a retinal image.

Zhou et al. [24] proposed a technique called Class Activation Mapping (CAM) to identify discriminative regions used by a restricted class in image classification. R. Selvaraju et al. [17] then introduced an advanced method called GradCAM based on the previous work. It generates visual explanations for any CNN-based network. The advantage of this method is that it doesn't require architectural changes or retraining.

Inspired by them, exploring perceptive interpretability based on DRIU network. We first followed [17], expecting to apply grad-cam, which is used to generate heat/saliency/relevance-map with the results of the final max-pooling layer, to our network. Soon we found it inappropriate because it is a more common approach in classification task while our work is a segmentation task. But we reserved the idea of plotting the saliency map since it seemed significant in the process of decision making even in a segmentation task. We then regarded the segmentation task as a particular case of classification task. We

Manuscript received October 5, 2024
Manuscript revised October 20, 2024
https://doi.org/10.22937/IJCSNS.2024.24.10.25

assumed when there are only two classes, classification will transform to segmentation. That assumption helped a lot in our interpretation. Besides, we employ this method on more layers such as convolution layers instead of just final maxpooling layer in order to explore more possibilities about interpretability.

A "Signal Method" referring to interpretability that observe the stimulation of neurons or a collection of neurons in [10] is also taken into consideration. After completion, there will be two advantages. On the one hand, the activated values of neurons can be manipulated for performance advancing. People can intervene the learning process of the network to guide it towards a right direction especially when the loss function is not perfect during all stages. On the other hand, this method offers another way to testify which part is most crucial in the network. The sensitivity of the stimulation value is highly related to the importance of that part.

2. Related Work

Our work focuses on the interpretability on retinal images. In that aspect, visualizing CNNs and medical explainability are essential.

A. Visualizing CNNs

Previous works [19, 23, 18, 6] have visualized CNN predictions by highlighting 'important' pixels. Some [4, 11] add an inversion of layers to explain the feature map. In those works, Guided Backpropagation [16] modify 'raw' gradients to get qualitative improvements. Most of these works are compared in [12]. Besides, in [15], LIME is another structure related to saliency method which may be instructive. Layer-wise Relevance Propagation (LRP) in [1] is also a method to construct saliency maps. If possible, these methods can be tuned for conducting saliency maps in segmentation task.

The most relevant to our approach is the Class Activation Mapping (CAM) approach in [24] and the improved Grad-CAM in [17]. They illustrated how every pixel contributes to the desired class in classification. We made some adjustments so that it can be used in segmentation tasks.

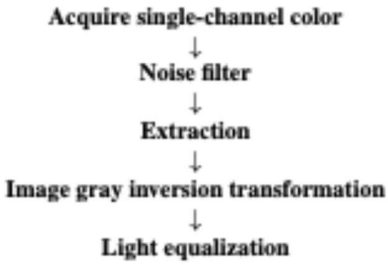
B. Interpretability

Most work in this part is summarized in [21, 2]. In these two papers, the authors introduced the concept of explainability, interpretability and many specific words in this field. They also reviewed many methods chasing for explainability in recent years.

3. Methodology

A. Preprocessing of data

The original retinal image has low contrast between the tissue target and the background, uneven illumination, and much noise. The original image needs to be processed in the following ways:



Step 1. Acquire single-channel color: Separate the original image into RGB three channels. The G (Green) channel has the largest contrast between the blood vessel and the background, so it is selected as the next input.

Step2. Noise filter: use Gaussian filter to eliminate isolated point noise, set r to 2 to avoid blurred images. The filter here is a two dimensional Gauss filter, which can be defined as.

$$G(x,y) = \frac{1}{2\pi\sigma^2} e \frac{x^2+y^2}{2\sigma^2} \qquad (1)$$

where σ represents the standard deviation, that is, the expansion width of the Gaussian function from the center, and x and y represent the distance from the center of the template in the x-axis and y-axis directions, respectively. Gaussian filtering is to change the pixel value of the pixel by the value of neighboring pixels and the corresponding weight. Therefore, the Gaussian filter is very likely to filter out noise and may also blur the image, mainly by setting the standard deviation σ. If σ is too large, the image is blurred. The purpose here is mainly to eliminate isolated noise without blurring the image, so σ only takes one pixel around the current pixel.

Step3. ROI extraction: To eliminate the influence of the background, we only process the region of interest, so we need to extract the ROI from the retinal touch image. Here we take retinal blood vessel segmentation as an example. The algorithm uses the entire fundus image as the processing object, so the ROI is the entire eyeball. This work uses the image and the mask extracts the ROI, that is, the image to be processed and the mask of the region of interest are multiplied bit by bit to obtain the ROI.

$$I_{ROI} = I_{filter} * Imask \quad (2)$$

Step4. Image gray inversion transformation: The blood vessel target in the retina image collected by the fundus camera is dark, and the rest of the retina is brighter than the blood vessel. To better adapt to human visual habits, the blood vessels and background colors are inverted here, that is, blood vessels are displayed in high brightness, and the rest of the retina is displayed in dark colors. The realization of the inverted transformation process is relatively simple. It only needs to subtract the pixel value of each pixel from the maximum value in the retinal single-channel image. The image needs to be normalized after filtering and ROI selection processing, so that the maximum value of pixels in the image is 1. Therefore, the normalization and gray inversion of the image are expressed as:

$$I_{ROI} = \frac{I_{ROI} - min(I_{ROI})}{max(I_{ROI}) - min(I_{ROI})}$$
$$I_{ROI} = 1 - I_{ROI}$$

Step5. Light equalization: Due to the uneven illumination of the fundus camera during the image collection process, and the uneven absorption and reflectivity of the light by each part of the retina, the contrast of each part in the retina image is uneven and the difference is large. To eliminate the influence of illumination on the algorithm's segmentation, the illumination equalization method is adopted for the retinal image:

$$I_{eq(x,y)} = I(x,y) + M - I_{W\_mean}(x,y)$$

where M represents the expected average pixel value, for 8-bit grayscale image, we directly take 128.

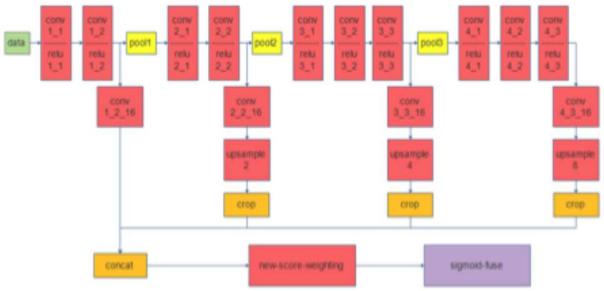


Fig. 1. DRIU Net structure

$I_{W\text{-}mean}(x,y)$ represents the average pixel value in window w.

B. *Deep Retinal Image Understanding (DRIU) Model*

The structure of the DRIU network is shown in Figure 1. and described as follows.

- ***Model Transformation***

In model training, we first transform the primitive caffe model into pytorch model to make it more convenient for later operations. We use the DRIU[14] network. It mainly consists of convolutional layers coupled with Rectified Linear Unit (ReLU) activations. The use of three max pooling layers in the architecture separates the network into four stages, each consisting of several convolutional layers. Between the pooling layers, feature maps of the same stage that are generated by convolutions with different filters have the same size. And we connect task-specific "specialized" convolutional layers to the final layer of each stage. Each specialized layer produces feature maps in K different channels, which are resized to the original image size and concatenated, creating a volume of fine-to-coarse feature maps. We append one last convolutional layer which linearly combines the feature maps from the volume created by the specialized layers into a regressed result. In our experiments, we used K = 16. Most convolutional layers employ 3 × 3 convolutional filters for efficiency, except the ones used for linearly combining the outputs (1 × 1 filters).

- ***Retraining***

First, we loaded the weights and bias from the caffe model and applied them into our pytorch model and then retrained the network. For training the network, we do not reproduce the class-balancing cross entropy loss function originally proposed in [22]

for the task of contour detection in natural images. Cause we just focus on the explanation of net work, we choose a base loss function of pytorch: Binary Cross Entropy Loss(BCELoss), which is used in binary classification. We denote the training dataset by S = (X_n, Y_n), n = 1,2,...,N with X_n being the input image and Y_n = y_j(n) , j= 1,...,X_n, y_j(n) ∈ {0, 1} the predicted pixel-wise labels. For i ∈ {1, 2, ..., N }, the BCELose function is defined as:

$$loss(x_i, y_i) = -w_i[y_i log x_i + (1 - y_i) log(1 - x_i)] \quad (3)$$

At training time, we fine-tune the entire architecture (base network and specialized layers) for 20000 iterations. We use Adaptive moment estimation with momentum, operating on one image per iteration. Due to the lack of data, the learning rate is set to a very small number (lr = 10-6), which is gradually decreased as the training process proceeds. We use several preprocessing methods in DRIU[14] and our methods mentioned before.

- ***Grad-CAM***

To understand whether the model focuses on blood vessels and where the focus of each layer of the neural network is, it is necessary to visualize each layer to see its impact. The CAM method requires that the model must use the GAP layer, because the GAP layer makes the model more interpretable. The idea is to use the weight w of the node as the weight of the feature map. The specific method is to select the node with the largest softmax layer value to propagate and calculate the gradient of the GAP layer as the weight of the feature map. However, the CAM method needs to change the network structure, which leads to the need to retrain the model, and the GAP layer is not suitable to appear in the blood vessel segmentation model. In order not to change the network structure of the model, we adopt the Grad-CAM method, which is characterized by not changing the structure of the model, but cleverly calculating the weight of each feature map. The principle of Grad-CAM is to select the node with the largest softmax value for back propagation, calculate the gradient of the layer we specify, and use the average value of each feature map as the weight of the feature map.

Step1: Calculate the partial derivative of the blood vessel probability yc corresponding to each pixel in the final output two-dimensional matrix with respect to all pixels A_{ij} of the specific layer feature map, where y is the probability value of each pixel in the output two-dimensional matrix, and c is the serial number of the class representing blood vessel. A is the feature map output by the specified layer, k is the serial number of the channel dimension of the feature map, i and j are the serial numbers of the width and height dimensions respectively.

$$\frac{\partial y_c}{\partial A_{ij}^k} \quad (4)$$

Step2: After calculating the partial derivative of y_c with respect to each pixel of the feature map, take a global average in the width and height dimensions.

$$\alpha_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y_c}{\partial A_{ij}^k} \quad (5)$$

Step3: Use the above results as weights to weight the feature maps of the specified layer, and finally combine them linearly. Then perform ReLU activation function processing on them.

$$L_{Grad-CAM}^c = RELU(\sum \alpha_k^c A^k) \quad (6)$$

- ***Mask:***

The image predicted by our model will generate much noise on the edge of both retinal region and the image which is not expected. As a result, we employed a mask operation before doing mIoU rating. This mask is generated from the original mask which is offered by the dataset. We reordered the mask at a one-pixel level so that the edge effect could be greatly reduced.

- ***Change Weights***

According to the heat map from Grad-CAM, we can find the contributions of each activations' outputs. Then we adjust the weights of our network. As we append one last convolutional layer which linearly combines the feature maps, we just need to change the weights of conv1 2 16, upsample2, 4 and 8 (according to Figure 1). To expand or decrease one feature map's influence, we multiply the feature map with a constant. And we use mIoU to test the performance of changed and unchanged results.

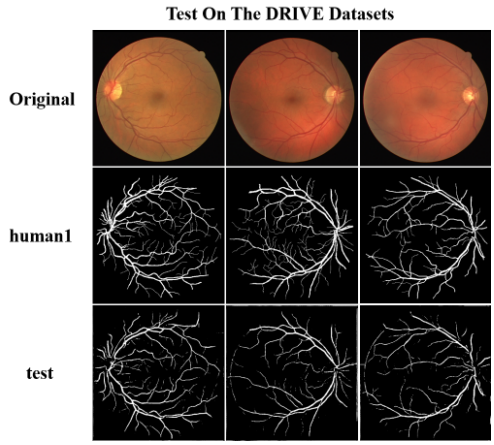


Fig. 2. The heat map on each layer

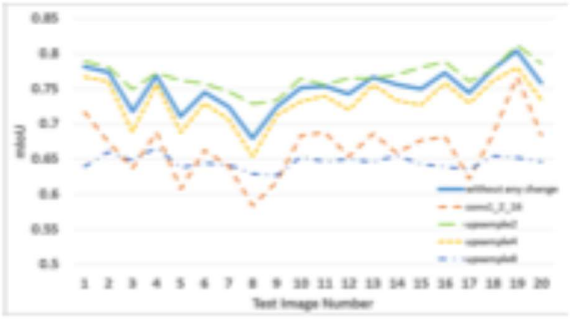


Fig. 3. MIoU after operations

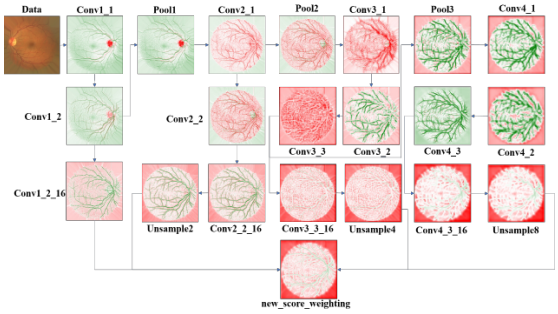


Fig. 4. The heat map on each layer

3. Experiments

A. Preprocessing of data

First, we perform preprocessing operations on the retinal image. This is the steps that most retinal blood vessel segmentation algorithms follow before image segmentation. It mainly includes acquire single-channel color, noise filter, ROI extraction, image grey inversion transformation and light equalization. Results are presented in Figure 2.

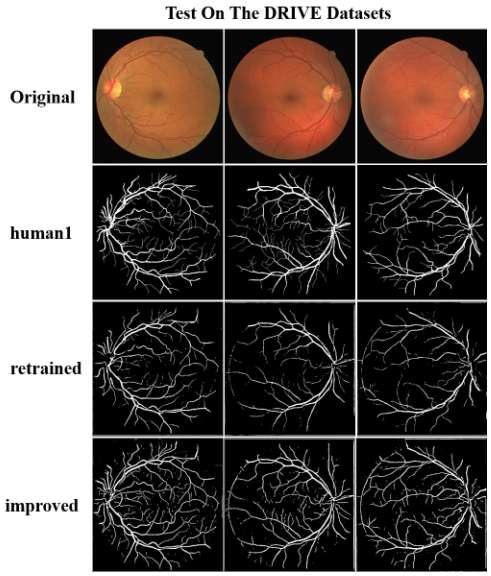


Fig 5. Blood vessels result on changed and unchanged network

B. Retraining Network

We tested the model on the DRIVE dataset. The initial results (presented in Figure 3) show that our network doesn't perform very well. This is understandable: we didn't use the class-balancing cross entropy loss function, and the layers of our network aren't pre-trained. It is obvious that our results have outlines. So, when we use mIoU on results, we do a mask on our photographs.

C. Grad-CAM

We use the Grad-CAM method for each layer to understand its impact on the results. Figure 3 show results of this step. The green part indicates a positive contribution to the blood vessel segmentation result, and the red part indicates a negative contribution to the result. All the layers focus on the blood vessel part, especially the upsample2 layer. And it shows a positive contribution in the blood vessel part. Therefore, we believe that the upsample2 layer could have a higher weight on the result. So, the accuracy of model segmentation can be improved by increasing the weight of the upsample2 layer.

- ***Change Weights***

According to the heat maps provided by Grad-CAM (Figure 4), we change the weights of layers

before the final convolutional layer. Figure 5 shows the results when we improve the upsample2 layer's weight, where the segmentations become better. The mIoU will be calculated for every image in the test set (only pixels within the mask are considered). We use the average mIoU as one of the basis for model evaluation, and the average mIoU of without any change is 0.7502. After improving the weight of upsample2 layer, the average mIoU is 0.7674, which is better than before. This corresponds with our hypothesis.

4. Conclusions and Discussions

According to our work, we thought there are two parts can be improved or continued to get more precise results. First, more datasets can be used for more precise interpretations. We only employed our method on DRIVE dataset due to many problems, most of which is in the retraining process. More datasets can help find more robust explanations. Besides, many other methods referred in the introduction part needed to be applied to segmentation tasks. Such trial may bring amazing results.

In this work, we put forward a series of pre-processing methods to raise the contrast of the retinal image. Although not used in later work, it is still a powerful approach according to the result image. Besides, we compute the saliency map on the segmentation task using grad-CAM, which is rarely done by others before. According to the saliency map, we adjusted the weights of some layers manually to explore the 'importance' further.

References

[1] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Mu¨ller, and W. Samek. On pixel-wise explanations for non- linear classifier decisions by layer-wise relevance propa gation. PloS one, 10(7):e0130140, 2015.

[2] A. Barredo Arrieta, N. D´ıaz-Rodr´ıguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera. Ex- plainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. Information Fusion, 58:82 – 115, 2020.

[3] L. Chen, P. Bentley, and D. Rueckert. Fully automatic acute ischemic lesion segmentation in dwi using convolutional neural networks. NeuroImage: Clinical, 15:633–643, 2017.

[4] A. Dosovitskiy and T. Brox. Inverting convolutional networks with convolutional networks. arXiv preprint arXiv:1506.02753, 4, 2015.

[5] M. T. Dzindolet, S. A. Peterson, R. A. Pomranky, L. G. Pierce, and H. P. Beck. The role of trust in automation reliance. International journal of human-computer studies, 58(6):697–718, 2003.

[6] C. Gan, N. Wang, Y. Yang, D.-Y. Yeung, and A. G. Hauptmann. Devnet: A deep event network for multimedia event detection and evidence recounting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.

[7] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.

[8] A. D. Hoover, V. Kouznetsova, and M. Goldbaum. Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response. IEEE Transactions on Medical Imaging, 19(3):203–210, 2000.

[9] C. J. Kelly, A. Karthikesalingam, M. Suleyman, G. Corrado, and D. King. Key challenges for delivering clinical impact with artificial intelligence. BMC medicine, 17(1):195, 2019.

[10] P.-J. Kindermans, S. Hooker, J. Adebayo, M. Alber, K. T. Schu¨tt, S. Da¨hne, D. Erhan, and B. Kim. The (un) reliability of saliency methods. arXiv preprint arXiv:1711.00867, 2017.

[11] A. Mahendran and A. Vedaldi. Understanding deep image representations by inverting them. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 5188–5196, 2015.

[12] A. Mahendran and A. Vedaldi. Salient deconvolutional networks. In B. Leibe, J. Matas, N. Sebe, and M. Welling, editors, Computer Vision – ECCV 2016, pages 120–135, Cham, 2016. Springer International Publishing.

[13] K.-K. Maninis, J. Pont-Tuset, P. Arbela´ez, and L. Van Gool. Deep retinal image understanding. In S. Ourselin, L. Joskowicz, M. R. Sabuncu, G. Unal, and W. Wells, editors, Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016, pages 140–148, Cham, 2016. Springer International Publishing.

[14] K.-K. Maninis, J. Pont-Tuset, P. Arbela´ez, and L. Van Gool. Deep retinal image understanding. In International conference on medical image computing and computer-assisted intervention, pages 140–148. Springer, 2016.

[15] M. T. Ribeiro, S. Singh, and C. Guestrin." why should i trust you?" explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD

international conference on knowledge discovery and data mining, pages 1135– 1144, 2016.
[16] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention, pages 234–241. Springer, 2015.
[17] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Oct 2017.
[18] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps, 2013.
[19] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Ried- miller. Striving for simplicity: The all convolutional net. arXiv preprint arXiv:1412.6806, 2014.
[20] J. Staal, M. D. Abra`moff, M. Niemeijer, M. A. Viergever, and B. Van Ginneken. Ridge-based vessel segmentation in color images of the retina. IEEE transactions on medical imaging, 23(4):501–509, 2004.
[21] E. Tjoa and C. Guan. A survey on explainable artificial intelligence (xai): towards medical xai. arXiv preprint arXiv:1907.07374, 2019.
[22] S. Xie and Z. Tu. Holistically-nested edge detection. In Proceedings of the IEEE international conference on computer vision, pages 1395–1403, 2015.
[23] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, Computer Vision – ECCV 2014, pages 818–833, Cham, 2014. Springer International Publishing.
[24] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.