IETI Transactions on Ergonomics and Safety

http://ieti.net/TES

2021, Volume 5, Issue 1, 24-35, DOI: 10.6722/TES.202105_5(1).0004

COMPARATIVE ANALYSIS OF METHODS FOR BLOOD VESSEL DETECTION IN RETINAL IMAGES

Radovan Turović[1,a], Gorana Gojić[1,b], Dinu Dragan[1,c], Veljko Petrović[1,d], Dušan Gajić[1,e] and Ana Oros[2,f]

[1]University of Novi Sad, Faculty of Technical Sciences, Novi Sad, Serbia

[2]Institute of Neonatology, Dept. of Retinopathy of Prematurity and Retinal Development, Belgrade, Serbia

[a]radovan.turovic@uns.ac.rs, [b]gorana.gojic@uns.ac.rs, [c]dinud@uns.ac.rs, [d]pveljko@uns.ac.rs, [e]dusan.gajic@uns.ac.rs, [f]annaoros03@yahoo.com

Abstract Images of the back part of the eye, also known as retinal images, are the basis for the diagnosis of many systemic and eye diseases such as glaucoma, diabetic retinopathy, and retinopathy of prematurity. Disease indicators may be found by observing the blood vessel network in the retinal image. A failure of an ophthalmologist to correctly identify a disease, due to fatigue or a low-quality image, may lead to severe health damage. To address this problem, many methods for automatic vessel detection in retinal images have been proposed. Among those, machine learning approaches based on convolutional neural networks have proven to yield the best results. Often, these methods require some sort of input retinal image preprocessing, such as transformation to grayscale, to emphasize blood vessels on images and reach their full potential. In this paper, we employ a subset of general-purpose algorithms for edge detection to produce retinal images with an emphasized retinal blood vessel network, which can be used for convolutional neural network blood vessel detection training. We test Canny, Sobel, Scharr, and Hollisticaly-Nested Edge Detection algorithms on the DRIVE dataset. Resulting images produced by these four algorithms are evaluated by an experienced ophthalmologist. Each image was graded and the time required to make the decision was measured. The ophthalmologist (who operated under double-blind test conditions) was later interviewed and qualitative data was collected. The data was then analyzed showing a clear win for the Sobel algorithm which, according to the post-test interview, preserves more fine detail.

Keywords: Computer vision; retinal images; blood vessel; edge detection.

1. INTRODUCTION

Medical image analysis is a crucial factor in the diagnosis and treatment of many diseases. By looking at the retinal image showing the back part of the eye, an ophthalmologist can assess some of widespread diseases such as glaucoma, diabetic retinopathy, and retinopathy of prematurity. To determine the diagnosis, the ophthalmologist observes the density and spatial distribution of the blood vessel network in the image. It can be a challenging task to precisely identify the complete blood vessel network due to a low and inconsistent contrast between finer blood vessels and the image background, blurred retinal images, ophthalmologist's fatigue, etc. A failure to correctly identify segments of the blood vessel network may result in severe eye damage due to disease progression, which eventually can lead to blindness. To help early disease diagnosis and treatment, many automated, image-based systems for blood vessel segmentation have been developed. These systems are intended to assist the ophthalmologist in blood vessel segmentation.

Early work in automatic blood vessel detection was based on classical methods for image analysis, such as hand-crafted filters [1–4] and line detectors [5,6]. However, recent state-of-the-art solutions

2021, Volume 5, Issue 1, 24-35, DOI: 10.6722/TES.202105_5(1).0004

for this task are based on deep learning and convolutional neural networks [7–9]. Given enough image data and corresponding outputs, a neural network can learn important features required to map an input image to the corresponding output image. Typically, neural networks for vessel detection receive a color retinal image as an input and, as a result, output a vessel segmentation mask or a probability map. If the input image is too complex or features that the network should learn are not expressive enough in the color image, the image is usually transformed to other, more expressive domains, and used as a standalone or an extra input to the neural network. In the particular case of blood vessel detection, the input color image is transformed to enhance blood vessels on the retinal image. Stationary Wavelet Transform (SWT) is used in [7] to transform the image to a grayscale space which is later used to train the convolutional neural network for blood vessel detection. In [8], the image is also converted into grayscale by extracting a green channel of the image, which is further enhanced by different filters.

The ability of the chosen transformation to emphasize image features that should be learned directly affects the performance of the neural network that is trained using transformed data. Although classic approaches for blood vessel detection are well behind deep learning approaches, they can still be used in a data preprocessing step, as is demonstrated by their common usage in existing methods. Inspired by this fact, we compare multiple well-established, general-purpose edge detection algorithms for automatic blood vessel enhancement on grayscale retinal image. The study is performed in a context of preprocessing transformations for retinal images used to train convolutional neural networks for automatic detection of a blood vessel network from a color image. Among classical edge detection algorithms, we choose Canny [10], Sobel [11], and Scharr [12] edge detectors since they are proven to be the most widely used of all non-machine-learning algorithms. We also include the Holistically-Nested Edge Detection (HED) [13] algorithm as a representative of general-purpose, deep learning algorithms for edge detection. All experiments are performed on images from DRIVE dataset [2]. We apply all four edge detection algorithms on each color retinal image to obtain images showing blood vessel edges. The results are then graded by an experienced ophthalmologist according to how faithfully they display the blood vessels visible on the corresponding color retinal image. In our experiment, we assume that the method with the best grades by the expert is the best preprocessing transformation among suggested due to the best emphasis on blood vessels.

The findings of this research are interesting not only for its application to retinal image analysis but as a way to analyze the geometric properties of edge-detection filters—what is preserved and what is lost and how this level of detail interacts with various requirements edge detection may pose.

The rest of the paper is organized as follows. In Section 2, we present an experimental setup used in this paper, including the retinal image dataset used and methods tested. Section 3 gives a brief theoretical overview of the methods used in this research. Next, in Section 4, we present the statistical and visual results of our experiment. Section 5 closes the paper with the main conclusions and plans for future work.

2. MATERIALS AND METHODS USED

The experiments are performed on retinal images from the DRIVE dataset. The dataset was collected as a part of a screening program conducted by the Netherlands and comprises of 40 color retinal

2021, Volume 5, Issue 1, 24-35, DOI: 10.6722/TES.202105_5(1).0004

images, including 7 images with pathology. All images are captured by a Canon CR5 nonmydriatic 3CCD camera at 45° field of view. For each image, a blood vessel network is manually annotated by three observers, including one expert in the field. The dataset is split into training and testing subsets, but since our experiment does not require training, all images are used for testing. Four characteristic images from the DRIVE dataset are shown in Figure 1.

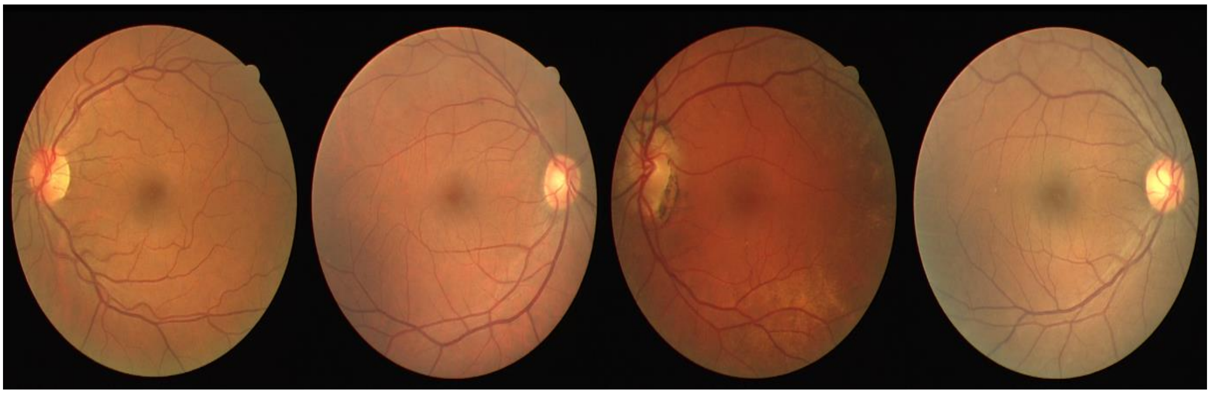

Figure 1. Examples of images used in the experiment from the DRIVE dataset.

The experiment was conducted on a workstation with the hardware specifications shown in Table 1. The operating system of the workstation is Ubuntu 18.04.3. All code used was written in the Python programming language, version 3.6 using version 4.8.2 of the Anaconda framework. The implementation of the Sobel, Scharr and Canny filters were the ones publicly available as part of the OpenCV library, version 3.4.2 through the Anaconda python wrapper.

Table 1. Hardware specification used in the experiment.

CPU	GPU	RAM
AMD Ryzen 7 1700	NVIDIA GeForce GTX TITAN X 11GB	32 GB

Every image in the DRIVE dataset was used during the experiment. The experiment was conducted in the following steps, also shown in Figure 2:

1. All the images that are used as the input for the algorithms tested are first preprocessed.

2. Preprocessed images are fed into the Sobel, Scharr, and Canny filters, and the HED neural network.

3. All the resulting images are collected and assembled into a survey. Observers are asked to grade how easy it is for them to evaluate the condition of the blood vessels based on the shown images.

4. The data acquired as part of the survey which included the grades and the precise time needed for grading was analyzed statistically.

2021, Volume 5, Issue 1, 24-35, DOI: 10.6722/TES.202105_5(1).0004

The concrete use of all the methods was based on the online documentation available for the OpenCV[1] library and the HED Git-hub repository[2] separately. For the classical methods, in all three cases, the first two steps were to smooth the image using a Gaussian filter in order to remove edge noise pixels and to convert the smoothed images from color to grayscale. Preprocessing for HED consists of simply turning images to grayscale to reduce color space complexity. From this step onwards, each method applies its own specific edge filter on the prepared input image. Parameters for each of the filters were selected independently. For Sobel, kernels of size 5x5 were used for calculations of second derivatives in both horizontal and vertical directions. In case of Scharr, for both directions first derivatives were used. Finally, Canny was used with lower and upper thresholds of 20 and 60. For Canny, thresholds were chosen based on the analysis of the histograms of grayscale images. Other parameters of these methods, if not listed, are left with their default values. For the HED neural network, the model used is the latest publicly available pretrained model[3]. Last step of all methods was to persist resulting edge-detected images.

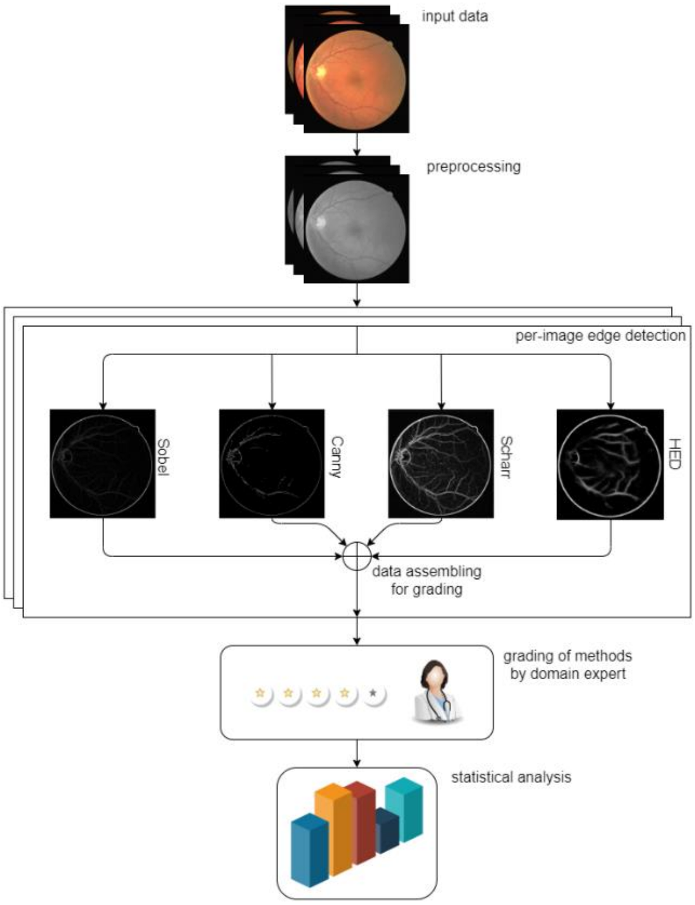

Figure 2. General workflow of the experiment.

[1] **https://docs.opencv.org/** [Accessed: 17 January 2020]

[2] **https://github.com/ashukid/hed-edge-detector** [Accessed: 17 January 2020]

[3] Referenced from **https://github.com/s9xie/hed** [Accessed: 17 January 2020]

2021, Volume 5, Issue 1, 24-35, DOI: 10.6722/TES.202105_5(1).0004

The survey includes the original images and the transformed images. Each question of the survey contains the original image, the transformed image, and the grading scale. We used a grading scale from 1 to 5, where one means that it was very hard and five means that it was very easy for them to evaluate the condition of the blood vessels based on the shown images. Example of a question from the survey is shown in Figure 3. Unfortunately, in this experiment, only one ophthalmologist participated, although in the future we plan to include more domain experts. The participating ophthalmologist is an expert in the field with particular expertise in the retinopathy of prematurity (ROP) and with more than 25 years of experience. The survey is in Serbian language for the benefit of the observer who is a native Serbian speaker. In the example in Figure 3, we did not translate the text, because we wanted to show the original survey in the form presented to the observer.

3. RESEARCH FUNDAMENTALS

Edge detection is a class of methods in Computer Vision that examine images for existence of edges. The core idea is to map every pixel to the probability that the pixel is a part of the edge. The edge is a border between two neighboring groups of pixels with a significant change in color or color intensity.

Generally speaking, there are two classes of edge detectors to consider:

- Classic straightforward methods, such as Sobel, Scharr and Canny.
- Machine learning methods, such as the HED neural network.

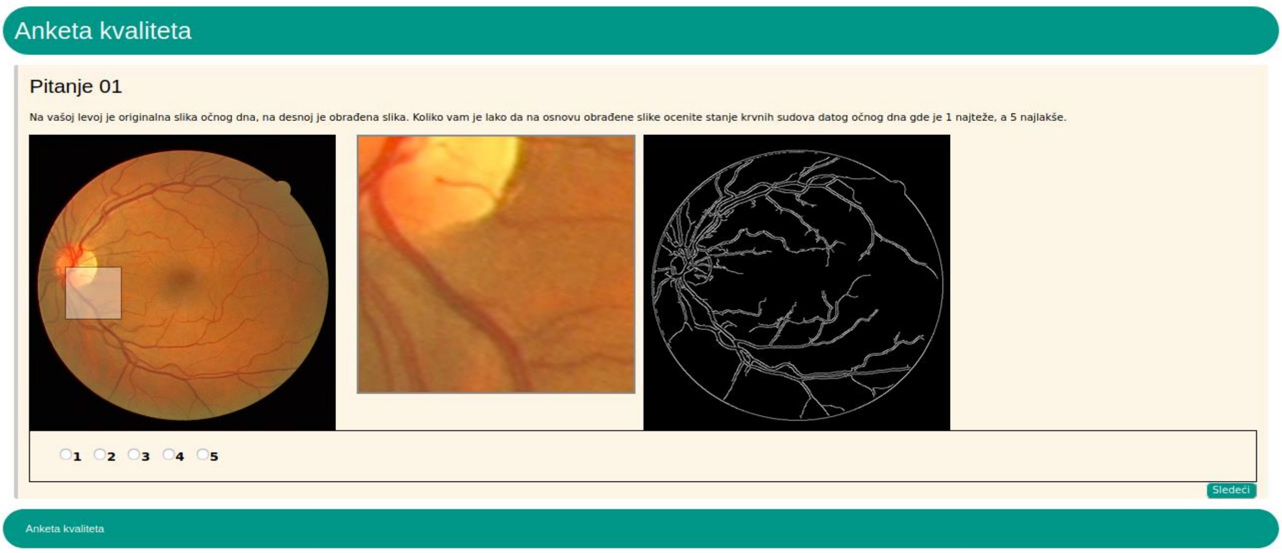

Figure 3. Example of a question used in the survey which the ophthalmologist had to answer (the question is in Serbian and has the following translation: “On the left side is the original retinal image and on the right side is the processed image. How easy it is for you to evaluate the condition of the blood vessels in the processed retinal image, where 1 means very hard and 5 means very easy”).

2021, Volume 5, Issue 1, 24-35, DOI: 10.6722/TES.202105_5(1).0004

The first class of methods proved themselves insufficient for detecting edges between objects with insufficient pixel differences in their edge regions. Consequently, various machine learning methods for edge detection in images were developed to tackle this problem. Since machine learning methods also look for the context of the image, it is common to train domain specific machine learning models. For example, in the case of edge detection described in this paper, there is a specific use-case of discerning blood vessels in retinal images.

Sobel [11], Scharr [12], and Canny [10] are based on a derivative analysis of an image. If an image is viewed as a discrete 2D function, whose inputs are the x and y coordinates of a pixel and whose output is the value of the pixel at [x, y]. An edge can then be modeled as a significant change in value of the function. Given that derivatives measure the rate of change of functions, they can be used to detect edges. In those regions of an image where edges are present, the derivative function reaches a local maximum, indicating that the relevant pixel belongs to the edge. To simplify calculation, the derivative is calculated from the partial derivatives over x and y, or in other words, in horizontal and vertical directions. Partial derivatives are calculated by means of convolution of each pixel with its neighboring pixels against a kernel as can be seen in Figure 4.

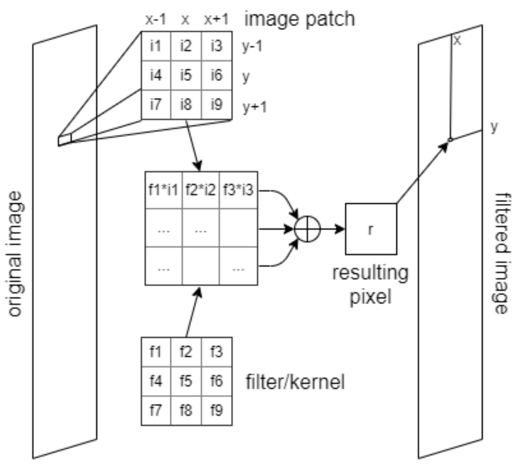

Figure 4. Filtering image by using convolution.

The kernel is an approximation of a derivative, enhanced with smoothing perpendicular to the direction of the derivative, so as to form a matrix. The total value of a derivatives indicates the probability of a given pixel belonging to the edge. The symmetrical Sobel kernels, used to implement the Sobel operator, can be seen in Figure 5. A linear combination of results after convolving a part of the image of the same size as the kernels, yields derivative value for a given pixel, i.e., edge probability.

a)

-1	*-2*	*-1*
0	*0*	*0*
1	*2*	*1*

b)

-1	*0*	*1*
-2	*0*	*2*
-1	*0*	*1*

Figure 5. 3-by-3 Sobel operator; a) derivative in y direction; b) derivative in x direction.

2021, Volume 5, Issue 1, 24-35, DOI: 10.6722/TES.202105_5(1).0004

Scharr in his thesis explored ways of optimizing kernels. His main contribution was rotational invariation of kernels. This practically means that kernels for edge detection should return the same probability without respect to angle at which an edge is observed in an image. The OpenCVs implementation of a 3-by-3 Scharr optimization of the Sobel operator can be seen in Figure 6.

a)

-3	*-10*	*-3*
0	*0*	*0*
3	*10*	*3*

b)

-3	*0*	*3*
-10	*0*	*10*
-3	*0*	*3*

Figure 6. 3-by-3 Scharr operator (OpenCV implementation); a) derivative in y direction; b) derivative in x direction.

Canny filter is an upgrade to Sobel filter with additional steps. Canny aims to emphasize the "significant" edges more, so in its two additional steps it removes insignificant edges and noise and assimilates edges that are in immediate proximity to each other. The first step is based on hysteresis. Using two thresholds, pixels are divided in three groups, as "not edges" if the computed probability is below the lower threshold, as "sure edges" if the probability is over the higher threshold, and the rest are labeled as potential edge pixels. Potential pixels are accepted as edges if there is a path of only potential pixels to one of the "sure edge" pixels. It should be noted that Canny edge detection is based on removing insignificant and duplicated edges. This removes smaller details from resulting image.

HED is a neural network for general purpose edge detection [13]. It is a deep fully convolutional neural network. Here deep means that there is a large number of layers and that the network has high computational and memory complexity. Fully convolutional means that from start to end, every layer is convolutional, excluding pooling and activation layers. Convolutional layers have images as inputs and outputs, and the output is calculated using filters/kernels as in Figure 4. Main difference as compared to simple filtering is that parameters of kernels are learned in the process of training the neural network. Concretely, HED tackles the problems of different scales of objects, complex feature learning, and comprehending an image as a whole. The HED architecture consists of stacked activation and pooling layers and mini-stacks of convolutional layers. HED outputs multiple images in various sizes each produced by a mini-stack of convolutional layers. The final output aggregates all the mini-stack outputs into a single image with the same dimensions as the input image.

4. RESULTS AND DISCUSSION

The first step of the analysis of the grading data is the computation of summary statistics and basic visualization. To this end, we produced the results visible in Table 2 and Figure 7. As it can be seen from Table 2 and Figure 7, the confidence intervals do not overlap indicating a high probability of statistically significant difference in the grade averages. Given the relatively small size of the sample, and the experimental design, it was decided that robust methods would be called for. In the end Wilcox's implementation of Yuen's method for comparing means [14–17] using pooled variance for repeated measures design with 10% trimmed means was employed with a *post-hoc* comparison design corrected using Holm's method. These results are presented in Table 3.

2021, Volume 5, Issue 1, 24-35, DOI: 10.6722/TES.202105_5(1).0004

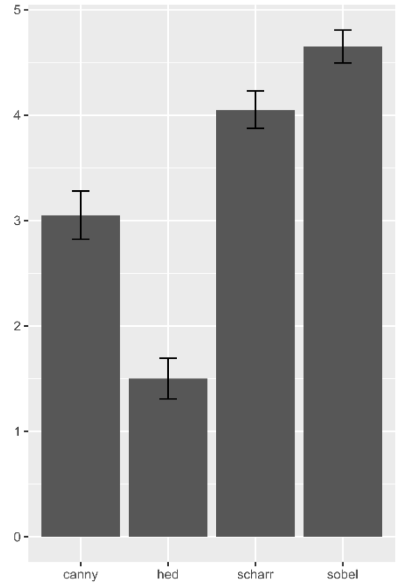

Figure 7. Mean plot of grades with 95% CIs, *n* = 40.

Table 2. Descriptive statistics of ophthalmologist grading data, *n* = 40.

Filter	Grade Mean ± 95% CI	Grade Standard Deviation
Canny	3.05 ± 0.22	0.71
Sobel	4.65 ± 0.15	0.48
Scharr	4.05 ± 0.17	0.55
HED	1.5 ± 0.19	0.60

When dealing with time, the methods were the same, albeit with more modest success. Figure 8 and Table 4 demonstrate the descriptive statistics done on the data.

Table 3. Result of group mean comparison, ophthalmologist grade, *n* = 40. A measure of zero for the probability indicates a probability value too small for the algorithm to detect.

Filter comparison	$\hat{\psi}$	$\hat{\psi}$ 95% CI lower bound	$\hat{\psi}$ 95% CI upper bound	p	Significant?
Scharr vs. Sobel	-3.190	-3.560	-2.820	0	Yes
HED vs. Sobel	-2.500	-2.780	-2.220	0	Yes
HED vs. Scharr	-1.590	-1.980	-1.210	3.15e-12	Yes
Canny vs. Sobel	-0.969	-1.400	-0.541	8.22e-7	Yes
Canny vs. HED	-0.625	-0.901	-0.349	8.22e-7	Yes
Canny vs. Scharr	1.590	1.20	1.99	5.16e-12	Yes

Given the partially overlapping confidence intervals and the unpromising nature of the stats, it is to be expected that the results of statistical analysis are similarly dubious. Indeed, as Table 5 shows, the results are unclear. While it is clear that the times are mostly different, it is impossible to differentiate between HED and Scharr and Canny and Sobel. In other words, there are two groups, the slow and the fast, but within those groups there is no difference that the statistical instruments used within this paper can detect.

2021, Volume 5, Issue 1, 24-35, DOI: 10.6722/TES.202105_5(1).0004

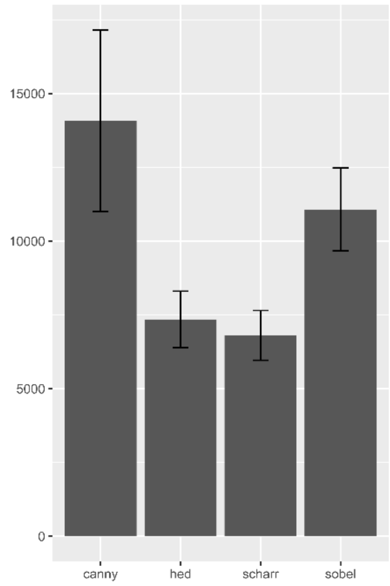

Figure 8. Mean plot of times with 95% CIs, *n* = 40. Time measured in ms.

Table 4. Descriptive statistics of ophthalmologist time data, *n* = 40.

Filter	Time Mean ± 95% CI	Time Standard Deviation
Canny	14080 ± 3067	9591
Sobel	11073 ± 1404	4390
Scharr	6808 ± 843	2635
HED	7350 ± 953	2981

Table 5. Result of group mean comparison, ophthalmologist time, *n* = 40.

Filter comparison	$\widehat{\psi}$	$\widehat{\psi}$ 95% CI lower bound	$\widehat{\psi}$ 95% CI upper bound	p	Significant?
Scharr vs. Sobel	-4329	-6711	-1947	9.11e-5	Yes
HED vs. Sobel	-3821	-6208	-1434	3.47e-4	Yes
HED vs. Scharr	452	-911	1815	3.99e-1	No
Canny vs. Sobel	1770	-2036	5575	3.99e-1	No
Canny vs. HED	4831	1680	7982	4.47e-4	Yes
Canny vs. Scharr	5631	2213	9048	2.98e-4	Yes

To visually illustrate the effects of the chosen methods on the retinal images, we provide Figure 9. The first row shows four randomly selected images from the DRIVE dataset, while the rows from the second to the fifth show corresponding result images produced by Canny, Sobel, Scharr, and HED methods, respectively.

5. CONCLUSION

From the presented experimental results, we can clearly conclude that, when it comes to quality, the Sobel edge detection is by far the best, followed by Scharr, and then trailed by Canny. HED is simply not worth using in this context. The results of the time comparison are less clear, but a possible interpretation of the results is that slower response times correspond to the level of detail, on the one hand, and the level of contrast, on the other. The experimental design did not allow us a way of

2021, Volume 5, Issue 1, 24-35, DOI: 10.6722/TES.202105_5(1).0004

determining if this interpretation is true. Therefore, more research is clearly required if this problem is to be solved.

The results of the grading are counterintuitive. Scharr is an upgrade of Sobel and yet is rated a little bit worse. Canny is an even further upgrade and is worse still. HED is a modern, machine-learning powered method, and yet it ranks worst of all. This is precisely the opposite of the expected results. Some additional context is accessible through the qualitative data gathered in a post-test interview with the ophthalmologist, expert. It turns out that the more advanced models that emphasize edges and clear out the noise also remove vital detail that the expert needs to make a diagnosis. The expert identified HED as completely lacking the small details needed for diagnostics (indeed, HED depicts only the broad outlines of an image: precisely what it has been designed to do), and Canny's thresholding approach creates what the expert called "empty areas", void of any texture or detail that is needed to characterize a retinal image. It should also be noted that HED, provided as-is, is a general-purpose edge detector CNN. There is a possibility that, should HED get explicitly trained for this use-case, it may yield better performance.

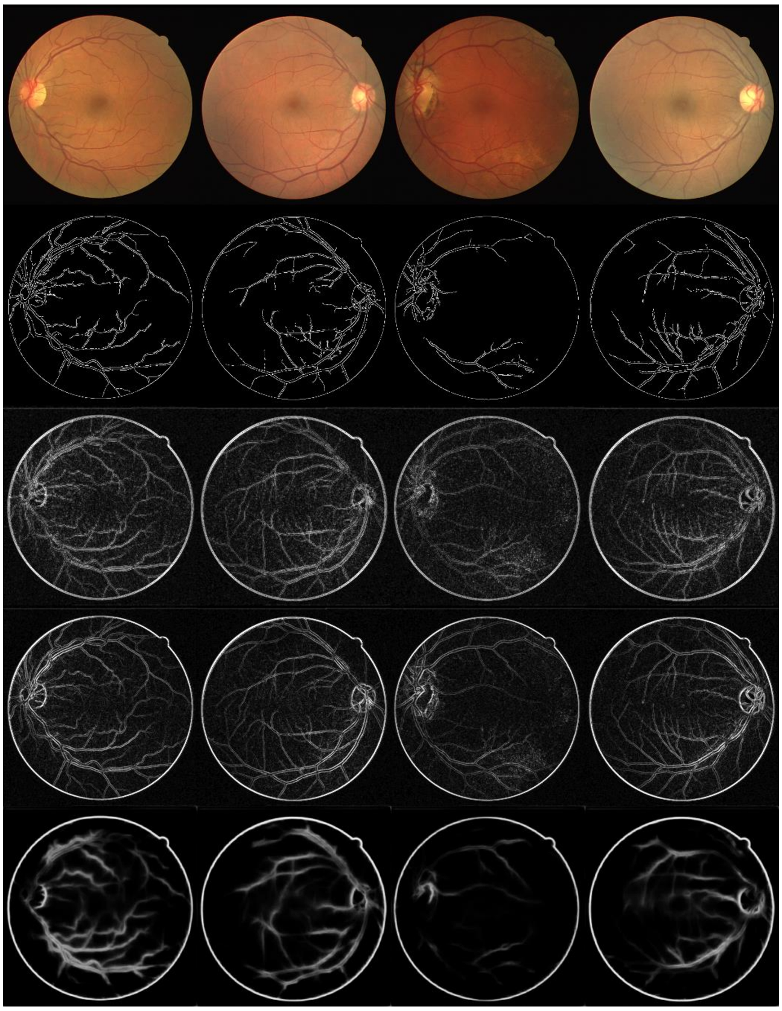

Figure 9. Original images (the first row) and resulting images (Canny, Sobel, Scharr and HED edge detectors respectively).

2021, Volume 5, Issue 1, 24-35, DOI: 10.6722/TES.202105_5(1).0004

From graphics standpoint pixel color and intensity values do not depict the difference between small blood vessels which form the region of interest from the background in a straightforward way. This renders classical edge detection methods unable to differentiate edges of the small blood vessels, while HED is unable to by design. With this fresh insight, we can conclude that general-purpose edge algorithms are not appropriate even for the pre-processing step. Methods which perform well for regular photographs are not suitable in the considered case. Filters used to prepare medical images of this type for machine learning model training need to preserve texture, small details, and to differentiate between thicker and thinner blood vessels. To this end, more research is needed in the construction of a custom-designed filter made to dramatically increase contrast and isolate precisely the features of interest in input images.

References

[1] Fraz M. M., Remagnino P., Hoppe A., Uyyanonvara B., Rudnicka A. R., Owen C. G., et al., 2012, Blood vessel segmentation methodologies in retinal images - A survey. *Comput Methods Programs Biomed.*, 108(1), pp. 407–33.

[2] Soares J. V. B., Leandro J. J. G., Cesar R. M., Jelinek H. F., Cree M. J., 2006, Retinal vessel segmentation using the 2-D Gabor wavelet and supervised classification, IEEE Trans Med Imaging, 25(9), pp. 1214–22.

[3] Fraz M. M., Remagnino P., Hoppe A., Uyyanonvara B., Owen C. G., Rudnicka A. R., et al., 2011, Retinal vessel extraction using first-order derivative of gaussian and morphological processing. Lect Notes Comput Sci Subser Lect Notes Artif Intell Lect Notes Bioinform, 6938 LNCS(PART 1), pp. 410–20.

[4] Budai A., Bock R., Maier A., Hornegger J., Michelson G., 2013, Robust vessel segmentation in fundus images, *Int J Biomed Imaging*,

[5] Ricci E., Perfetti R., 2007, Retinal blood vessel segmentation using line operators and support vector classification, *IEEE Trans Med Imaging*, 26(10), pp. 1357–65.

[6] Nguyen U. T. V., Bhuiyan A., Park L. A. F., Ramamohanarao K., 2013, An effective retinal blood vessel segmentation method using multi-scale line detection, *Pattern Recognit*., 46(3), pp. 703–15

[7] Oliveira A., Pereira S., Silva C. A., 2018, Retinal vessel segmentation based on Fully Convolutional Neural Networks, *Expert Syst Appl*., pp. 112:229–42.

[8] Leopold H. A., Orchard J., Zelek J. S., Lakshminarayanan V., 2019, PixelBNN: Augmenting the Pixelcnn with batch normalization and the presentation of a fast architecture for retinal vessel segmentation, *J Imaging.*, 5(2)

[9] Maninis K. K., Pont-Tuset J., Arbeláez P., Van Gool L., Deep retinal image understanding. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), *Springer Verlag*, pp. 140–8

[10] Canny J., 1986, A Computational Approach to Edge Detection, *IEEE Trans Pattern Anal Mach Intell.*, PAMI-8(6), pp. 679–98

[11] Kanopoulos N., Vasanthavada N., Baker R. L., 1988, Design of an Image Edge Detection Filter Using the Sobel Operator, *IEEE J Solid-State Circuits*, 23(2), pp. 358–67

[12] Scharr H., 2000, Optimale Operatoren in der Digitalen Bildverarbeitung, University of Heidelberg, Germany

[13] Xie S., Tu Z., 2015, Holistically-nested edge detection, *Proc IEEE Int Conf Comput Vis.*, 2015 Inter, pp. 1395–403.

[14] Wilcox R. R., 1977, Pairwise Comparisons Using Trimmed Means or M-Estimators When Working with Dependent Groups, *Biom J.*, 39(6), pp. 677–88

[15] Wilcox R. R., 2012, Introduction to robust estimation and hypothesis testing

2021, Volume 5, Issue 1, 24-35, DOI: 10.6722/TES.202105_5(1).0004

[16] Wilcox R. R., Tian T. S., 2011, Measuring effect size: A robust heteroscedastic approach for two or more groups, *J Appl Stat*, 38(7), pp.1359–68.
[17] Mair P., Schoenbrodt F., Wilcox R.., 2014, WRS2: Wilcox robust estimation and testing